TALLINN UNIVERSITY OF TECHNOLOGY
School of Information Technologies

Anton Vishnevski 242775IVCM

MATRIX- Multi Agent Twin for Reality, Influence, and X-domain

Master's thesis

Supervisor: Pavel Chikul
MSc.

Co-Supervisor: Ricardo Gregorio Lugo
PhD.

Tallinn 2026

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Anton Vishnevski 242775IVCM

MATRIX- Mitmeagendiline kaksiksüsteem reaalsuse, mõjutamise ja X-domeeni jaoks

Magistritöö

Juhendaja: Pavel Chikul

MSc.

Kaasjuhendaja: Ricardo Gregorio Lugo

PhD.

Tallinn 2026

Author’s declaration of originality

I hereby certify that I am the sole author of this thesis. All the used materials, references to the literature and the work of others have been referred to. This thesis has not been presented for examination anywhere else.

Author: Anton Vishnevski

16.05.2026

Tallinn 2026

Abstract

This thesis presents MATRIX (Multi Agent Twin for Reality, Influence, and X-Domain), a multi-agent simulation framework designed to model influence operations and social interactions using large language model (LLM) agents. The research explores whether LLM-based agents can generate believable, context-aware content that adapts to specific online communities while amplifying selected behavioral and ideological traits. Reddit was chosen as the primary experimental environment due to its structured communities, conversational threads, and measurable engagement signals.

The proposed architecture combines subreddit-adaptive agents, trait-conditioning mechanisms, automated judge models, and conversation-chain analysis into a unified simulation pipeline. Parameter-efficient fine-tuning (PEFT) was used to train subreddit-specific adapters capable of reproducing community-specific language and interaction styles. Additional trait adapters were developed to amplify characteristics such as sentiment, toxicity, skepticism, and argumentative behavior. Automated evaluation models were then used to assess realism, coherence, and engagement potential of generated content.

Experimental results show that adapted agents outperform baseline models in community alignment and contextual realism, while trait amplification remains controllable without degrading conversational plausibility. The thesis contributes a modular framework for studying influence operations, synthetic social environments, and emergent multi-agent behavior in online ecosystems.

This thesis is written in English and is 135 pages long, including 11 chapters, 44 figures and 12 tables.

Lühikokkuvõte

MATRIX- Mitmeagendiline kaksiksüsteem reaalsuse, mõjutamise ja X-domeeni jaoks

Käesolev magistritöö esitleb raamistikku MATRIX, mis on suurtele keelemudelitele (LLM) põhinev mitmeagendiline simulatsioonikeskkond mõjuoperatsioonide ja veebisuhtluse modelleerimiseks. Uurimistöö eesmärk on hinnata, kas LLM-põhised agendid suudavad genereerida usutavat ja kontekstiteadlikku sisu, mis kohandub erinevate veebikogukondadega ning võimaldab võimendada valitud käitumuslikke ja ideoloogilisi tunnuseid. Peamiseks eksperimentaalseks keskkonnaks valiti Reddit selle kogukondliku struktuuri, aruteluahelate ja mõõdetavate kaasatusnäitajate tõttu.

Pakutud arhitektuur ühendab subreddit-spetsiifilised agendid, tunnuspõhise kohandamise mehhanismid, automaatsed hindamismudelid ning vestlusahelate analüüsi ühtseks simulatsioonitoruks. Subredditidele kohandatud adapterite treenimiseks kasutati parameetritõhusat peenhäälestamist (PEFT), mis võimaldas mudelitel omandada konkreetsetele kogukondadele iseloomuliku keelekasutuse ja suhtlusstiili. Lisaks loodi tunnusadapterid omaduste nagu sentiment, toksilisus, skeptitsism ja argumentatiivsus võimendamiseks. Genereeritud sisu realismi, sidusust ja kaasamispotentsiaali hinnati automaatsete hindamismudelite abil.

Eksperimentide tulemused näitavad, et kohandatud agendid ületavad baasmodelle kogukondliku sobivuse ja kontekstuaalse realismi osas, säilitades samal ajal kontrollitava tunnuste võimendamise ilma vestluse usutavust täielikult kaotamata. Töö peamine panus on modulaarse raamistiku loomine mõjuoperatsioonide, sünteetiliste sotsiaalsete keskkondade ja emergentse mitmeagendilise käitumise uurimiseks veebikeskkondades.

Lõputöö on kirjutatud inglise keeles ning sisaldab teksti 135 leheküljel, 11 peatükki, 44 joonist, 12 tabelit.

List of abbreviations and terms

ABM	Agent-based modeling
Adapter	Lightweight trainable module attached to a pretrained neural network to specialize model behavior for a specific task, domain, or behavioral characteristic without modifying the full base model
Agent	An adapted language model participating in simulations
AI	Artificial Intelligence
API	Application Programming Interface
AUC	Area Under the Receiver Operating Characteristic (ROC) Curve; evaluation metric measuring how well a model separates positive and negative classes across different decision thresholds
BART	Bidirectional and Auto-Regressive Transformers; transformer-based sequence-to-sequence model developed by Meta for natural language understanding and generation tasks
BERT	Bidirectional Encoder Representations from Transformers; transformer-based language model architecture designed for contextual understanding of text through bidirectional attention
BERTopic	Topic-modeling framework combining transformer-based embeddings, dimensionality reduction, and clustering methods to discover semantically coherent topics in text corpora
Big Five	Psychological personality framework describing human behavioral tendencies through five major dimensions: openness, conscientiousness, extraversion, agreeableness, and neuroticism
Bot	Automated online actor used to simulate human-like participation in digital communication environments, including posting, replying, amplification, or coordinated interaction behaviors
Collective Behaviour	Group-level interaction dynamics emerging from coordinated or repeated interactions between agents inside a shared communication environment
Corpus / Corpora	Large structured collections of textual or linguistic data used for analysis, training, validation, or evaluation in natural language processing and computational research

Decoding-Time Intervention	Modification or control of language model output during token generation, including reranking, filtering, logit adjustment, constrained decoding, or sampling control mechanisms
DISARM	an open-source framework for analyzing and countering disinformation and influence operations using a structured taxonomy of tactics, techniques, and procedures
Embedding	Dense numerical vector representation of text, tokens, or other data objects that captures semantic or contextual relationships in a continuous vector space
Feature	Measurable attribute or derived variable used as input for machine learning, statistical analysis, or model evaluation
Fine-Tuning	Process of further training a pretrained language model on task-specific or domain-specific data to adapt its behavior, style, or capabilities
GPU	Graphics Processing Unit
Graph	Mathematical structure consisting of nodes and edges used to represent relationships, interactions, or connectivity between entities
HDBSCAN	Hierarchical Density-Based Spatial Clustering of Applications with Noise; clustering algorithm that groups data points based on density while identifying noise and outliers
Hugging Face	Open-source machine learning platform and ecosystem providing pretrained models, datasets, libraries, and tools for natural language processing and generative AI research
Influence campaign	Coordinated sequence of influence operations conducted over time with the objective of shaping public perception, discourse, decision-making, or behavior within a target audience or online community
Influence operation	Coordinated information activity intended to shape public perception, opinion, or behavior within digital information environments
InstructGPT	Large language model framework introduced by OpenAI that uses supervised fine-tuning and reinforcement learning from human feedback to improve instruction-following behavior
Instruction-Tuned Model	Large language model additionally trained to follow human instructions, conversational prompts, and task-oriented requests more reliably
Jaccard Similarity	Statistical similarity measure comparing the overlap between two sets by dividing the size of their intersection by the size of their union

Judge	Evaluation model used to estimate the quality, contextual fit, community acceptance, or preference ranking of generated textual content
Kendall Correlation (Kendall's Tau)	Rank-based statistical measure evaluating the consistency of ordering between two ranked variables
Latent Space	Abstract multidimensional representation space in which semantically or behaviorally similar data points are positioned closer together by a machine learning model
Leiden Algorithm	Graph community-detection algorithm improving upon Louvain by producing better-connected and more stable clusters during modularity optimization
LIWC	Linguistic Inquiry and Word Count; text-analysis methodology and software framework used to estimate psychological, emotional, cognitive, and social characteristics from language patterns
LLM	Large Language Model
LoRA	Low-Rank Adaptation
Louvain Algorithm	Community-detection algorithm for networks and graphs that identifies clusters by optimizing modularity between connected nodes
MATRIX	Multi Agent Twin for Reality, Influence, and X-Domain
ML	Machine Learning
MNLI	Multi-Genre Natural Language Inference; benchmark dataset used for training and evaluating models on textual entailment, contradiction, and neutral relationship detection
MongoDB	Document-oriented NoSQL database management system used for storing and querying large-scale semi-structured data collections
MOSAIC	Multi-Agent Social Simulation framework based on large language model agents, designed for studying social interaction, coordination, and emergent behavior within simulated environments
NLP	Natural Language Processing
NSFW	Not Safe For Work
OASIS	Open Agent Social Interaction Simulations
OCEAN	Acronym for the five-factor personality model dimensions: Openness, Conscientiousness, Extraversion, Agreeableness, and Neuroticism
OpenAI	Artificial intelligence research and deployment organization known for developing large language models, generative AI systems, and reinforcement learning frameworks

Pairwise AUC	Ranking-oriented AUC metric measuring how often a model correctly orders pairs of samples according to preference or target score
PEFT	Parameter-Efficient Fine-Tuning; family of methods that adapt large language models by training only a small subset of additional parameters instead of updating the full model
Persona	Operational behavioral profile representing recurring linguistic, emotional, rhetorical, or ideological characteristics associated with an agent or communication style
Polarization	Process in which opinions, attitudes, or social groups become increasingly divided into opposing positions, often accompanied by reduced mutual agreement and increased ideological or emotional separation
Prompt	Structured textual input provided to a language model to guide generation behavior, contextual interpretation, or task execution
Prompt Conditioning	Technique for steering language model behavior through structured prompts, instructions, examples, or contextual information provided at inference time
Propaganda	Deliberately constructed communication intended to influence perceptions, beliefs, emotions, or behavior of an audience, often through selective presentation, emotional framing, repetition, or manipulation of information
Python	High-level general-purpose programming language widely used in machine learning, data analysis, automation, and scientific computing
QLoRA	Quantized Low-Rank Adaptation; parameter-efficient fine-tuning method combining low-rank adapters with low-bit quantized backbone models to reduce memory usage during training
Qwen	Family of large language models developed by Alibaba Cloud, supporting instruction-following, multilingual processing, and generative language tasks
RAM	Random Access Memory
RangeLab	Experimental laboratory environment within MATRIX used for controlled simulation, visualization, and evaluation of multi-agent conversational and influence-oriented interactions
Ranking	Process of ordering items according to predicted relevance, preference, similarity, or importance based on a scoring function or model output
Recall	Classification metric measuring the proportion of actual positive samples correctly identified by a model

Reddit	Social media and discussion platform organized into topic-specific communities called subreddits, where users create posts, comments, and threaded discussions evaluated through voting mechanisms
Regression	Statistical and machine learning method used to predict continuous numerical values from input features or representations
Regression	Statistical and machine learning approach used to predict continuous numerical values from input data
Reinforcement Learning	Machine learning paradigm in which a model learns behavior by optimizing actions according to reward signals derived from an environment, evaluator, or preference model
Reranking	Multi-stage evaluation process where preliminary candidates are rescored and reordered using additional contextual or preference-aware models
Ridge Regression	Regularized linear regression method using L2 penalty to reduce overfitting and stabilize coefficient estimation under multicollinearity
RMSE	Root Mean Squared Error; regression metric measuring the average magnitude of prediction errors between predicted and actual values
RQ	Research Question
SBERT	Sentence-BERT; modification of BERT optimized for generating semantically meaningful sentence embeddings suitable for similarity and clustering tasks
Semantics	Linguistic and contextual meaning expressed by text, including concepts, intent, relationships, and implied interpretation
Sentiment	Emotional polarity or affective orientation expressed in text, typically represented along dimensions such as positive, negative, or neutral attitude
Spam	Unsolicited, repetitive, or low-value content distributed in large quantities across digital platforms, often intended for promotion, manipulation, or disruption
Spearman Correlation	Non-parametric statistical measure evaluating the monotonic relationship between two ranked variables
Syntax	Structural organization and grammatical arrangement of words, symbols, and sentences within language
TF-IDF	Term Frequency – Inverse Document Frequency; statistical text representation method measuring the importance of words relative to a document collection

Trait	Operationally defined textual or behavioral signal, such as sentiment, politeness, toxicity, skepticism, or emotional tone, used for analysis and controllable agent specialization
Transformer	Neural network architecture based on self-attention mechanisms, widely used in modern natural language processing and large language models
T-SNE	t-Distributed Stochastic Neighbor Embedding; nonlinear dimensionality-reduction algorithm commonly used for visualization of high-dimensional embeddings
UMAP	Uniform Manifold Approximation and Projection; dimensionality-reduction technique used for visualization and clustering of high-dimensional data
VRAM	Video Random Access Memory

Table of contents

List of figures 15

List of tables 17

1 Introduction 18

1.1 Problem Statement and Contribution 19

2 Related Work 21

2.1 Influence Operations and Disinformation 21

2.2 Agent-Based Social Simulation 22

2.3 Controllable Text Generation 24

2.4 Trait and Style Modeling 24

3 System Design and Pipeline 28

3.1 High-Level Architecture 28

3.2 Data Sources 30

3.3 Design Principles 31

3.3.1 Separation of Storage and Computation 31

3.3.2 Reproducibility 31

3.3.3 Modularity 32

3.3.4 Parameter-Efficient Adaptation 32

3.3.5 Data Usage and Legal Consideration 32

3.3.6 Offline Simulation 33

4 Community Discovery and Dataset Preparation 34

4.1 Subreddit Discovery 36

4.2 Community Filtering 38

4.3 Data Extraction 40

4.4 Embedding and Visualization 42
5 Judge Models for Community Context 44
5.1 Feature Engineering 45
5.2 Model Training 46
5.3 Validation of Judges 47
6 Multi-Agent Training 51
6.1 Base LLM and Adaptation 52
6.2 Training Agents 56
6.3 Model Merging 61
7 Conversation Chain Construction and Trait Extraction 63
7.1 Building Conversation Graph 63
7.2 Trait/Persona Signals 67
7.3 Chain-Level Analysis 70
7.4 Correlation Analysis 73
8 Trait-Conditioned Agent Amplification 79
8.1 Trait-Adaptive Training 80
8.2 Simulation of Amplified Agents 82
9 Evaluation and Validation 85
9.1 Automated Metrics 86
9.2 Experimental Setup 88
9.3 Results and Analysis 90
10 MATRIX RangeLab: Interactive Multi-Agent Simulation Environment 95
10.1 Design Goals and Motivation 95
10.2 System Architecture of RangeLab 97
10.3 Conversation Tree Construction and Branching 99
10.4 Trait-Aware Candidate Generation and Judge-Guided Selection 101
10.5 Interactive Trait Visualization and Behavioral Monitoring 104

10.6 Experimental Simulation Scenarios 106
10.7 Limitations of the Simulation Environment 108
11 Discussion and Implications 110
11.1 Summary of Main Findings 110
11.2 Judge Models as Community-Preference Estimators 113
11.3 Community Adaptation and Influence Realism 115
11.4 Trait Amplification and Behavioral Drift 116
11.5 Trade-Off Between Control and Authenticity 118
11.6 Prompt Engineering Versus Adapter-Based Conditioning 120
11.7 Autonomous Risks and Cybersecurity Implications 121
11.8 Limitations and Future Work 122
11.9 Conclusion of the Discussion 123
References 125
Appendix 1 – Non-exclusive licence for reproduction and publication of a graduation thesis 135

List of figures

Figure 1. Seven strategies based on DISARM techniques for influence operations in social networks 22
Figure 2. Structure of MOSAIC 23
Figure 3. Structure of OASIS 23
Figure 4. Big Five OCEAN 26
Figure 5. Processing pipeline 29
Figure 6. Data workflow 35
Figure 7. Sample of minimum schema document 41
Figure 8. UMAP visualization 43
Figure 9. Distribution of pairwise AUC across subreddits 47
Figure 10. Scatter plot of Spearman correlation vs. pairwise AUC across subreddits 48
Figure 11. Bar chart comparing top-1 selection accuracy of judge model vs. random baseline 49
Figure 12. Multi-Adapter architecture 52
Figure 13. PEFT adaptation schematic 56
Figure 14. Deltas over standard model 60
Figure 15. Win rates among subreddits 60
Figure 16. Sample of chainmix unit 64
Figure 17. Sample of analyzed chain 72
Figure 18. Top negative features, Spearman 75
Figure 19. Top positive features, Spearman 76
Figure 20. Top ridge weights 77
Figure 21. AskReddit top/bottom Pearson 77
Figure 22. AskReddit top/bottom Ridge 78
Figure 23. AskReddit top/bottom Spearman 78
Figure 24. Per subreddit trait delta comparison sample 92
Figure 25. Per subreddit tournament trait results sample 93
Figure 26. Per subreddit judged delta comparison sample 93
Figure 27. Per subreddit judged tournament results sample 94

Figure 28. Main user interface of MATRIX RangeLab 99
Figure 29. Example synthetic discussion tree with multiple generated branches 101
Figure 30. Sample of candidate tournament table 103
Figure 31. Sample of scatter visualization of candidate trait alignment and judge scores 104
Figure 32. Sample of real-time thread-level trait progression visualization 105
Figure 33. Sample of trait heatmap across generated conversation branches 106
Figure 34. Sample of per-message trait analysis panel 106
Figure 35. Sample of trait dynamics with base vs politeness traits 108
Figure 36. Same trait dynamic presented by heatmap 108
Figure 37. High-level operational flow of MATRIX 112
Figure 38. Spearman correlation vs pairwise AUC 113
Figure 39. Distribution of pairwise AUC across subreddits 114
Figure 40. Delta of scores between base model and lightly specialized one 115
Figure 41. Spearman correlation between scores and traits in AskReddit 116
Figure 42. Heatmap of overall trait changes during amplification of selected trait 117
Figure 43. Negative score effect of specialization in the NBA subreddit 119
Figure 44. Specialization in politics subreddit 119

List of tables

Table 1. Storage and compute strategy 30
Table 2. Model adaptation strategies 32
Table 3. Community discovery strategies 37
Table 4. Filtering and validation methods 39
Table 5. Data sources and representation layers 41
Table 6. Feature families in judge models 45
Table 7. Judge model objectives 46
Table 8. Aggregate evaluation results 49
Table 9. Candidate base models for MATRIX agents 53
Table 10. PEFT methods 54
Table 11. Preliminary evaluation 59
Table 12. Trait and persona signal families 69

1 Introduction

Over the past decade, coordinated disinformation campaigns have evolved from manually orchestrated efforts into highly scalable, semi-automated operations. State and non-state actors increasingly exploit online platforms to shape narratives, influence public opinion, and destabilize information ecosystems [1], [2]. Incidents surrounding elections, geopolitical conflicts, and public health crises have demonstrated how rapidly such campaigns can propagate and adapt.

Recent studies highlight the emerging role of large language models (LLMs) and autonomous agents in amplifying these operations. Research such as the University of Southern California study [3] suggests that AI-driven agents are becoming capable of generating persuasive, context-aware content at scale, lowering the barrier to entry for influence operations.

Despite this evolution, there remains a critical gap: the lack of realistic, controllable simulation environments for studying such campaigns. Existing tools often focus either on network diffusion (like OASIS from CamelAI [4]) or isolated text generation, but fail to capture the interplay between agent behavior, stylistic traits, and community context [5].

This thesis adopts the concepts of information operations and influence campaigns as structured efforts to manipulate perception and behavior within a target population through coordinated messaging strategies. These operations increasingly rely on subtle stylistic adaptation, stance, emotionality, rather than overt propaganda.

To study this phenomenon, the platform Reddit is selected as the primary data source. Reddit provides a uniquely suitable environment due to:

1. it’s clearly segmented community structure (subreddits),
2. publicly accessible large-scale data,
3. rich conversational threads with contextual dependencies,
4. measurable feedback signals (e.g. upvotes, replies).

These properties allow for modeling both local conversational dynamics and global behavioral patterns, which are essential for simulating realistic influence campaigns.

1.1 Problem Statement and Contribution

The central problem addressed in this thesis is the absence of a unified framework for simulating influence campaigns using AI agents that exhibit controlled behavioral traits within realistic social contexts.

The MATRIX system is proposed as a solution: a simulation framework that trains large language model-based agents to generate context-aware content while systematically amplifying specific stylistic, emotional, and ideological traits.

The core research questions are formulated as follows:

RQ1: Can LLM-based agents be trained to generate contextually coherent content within specific online communities?

RQ2: To what extent can textual traits (e.g., sentiment, stance, toxicity, persuasion style) be learned, controlled, and amplified in generated outputs?

RQ3: To what extent such system could be developed without human validation and interaction?

To address these questions, the thesis pursues the following objectives:

1. design and implement a scalable data processing pipeline for large-scale Reddit data (≈2TB), following a compute-in-Python paradigm,
2. develop a trait extraction framework combining zero-shot classification and regression-based stabilization,
3. train subreddit-specific LLM agents using parameter-efficient fine-tuning (LoRA),
4. construct evaluation mechanisms, including judge models for estimating content effectiveness,
5. build a prototype simulation environment where multiple agents interact within reconstructed conversation chains,
6. analyze the ability of agents to replicate and amplify influence strategies.

The primary deliverables include trained agent models, trait representations, evaluation metrics, and a working prototype of the MATRIX simulation system.

2 Related Work

This chapter reviews existing research on influence operations, computational propaganda, agent-based social simulation, controllable text generation, and trait-based modeling. It establishes the theoretical foundation and identifies the gap addressed by the MATRIX framework.

2.1 Influence Operations and Disinformation

Influence operations (IO) and computational propaganda have become central topics in the study of modern information ecosystems. These operations are typically defined as coordinated efforts to manipulate public perception and behavior through strategic dissemination of information [6]. Early research focused on the role of social bots and coordinated human actors in amplifying narratives across platforms, particularly during political events and crises [7].

Recent work highlights a transition toward AI-assisted and AI-driven influence campaigns. Pastor-Galindo et al. (2025) [8] propose a taxonomy of seven core strategies in influence operations, including narrative release, amplification, information pollution, and targeted degradation of opposing viewpoints (Figure 1). Strategies are mapped to DISARM framework [9]. These strategies emphasize that modern disinformation campaigns rely less on overt propaganda and more on subtle manipulation of tone, framing, and emotional appeal.

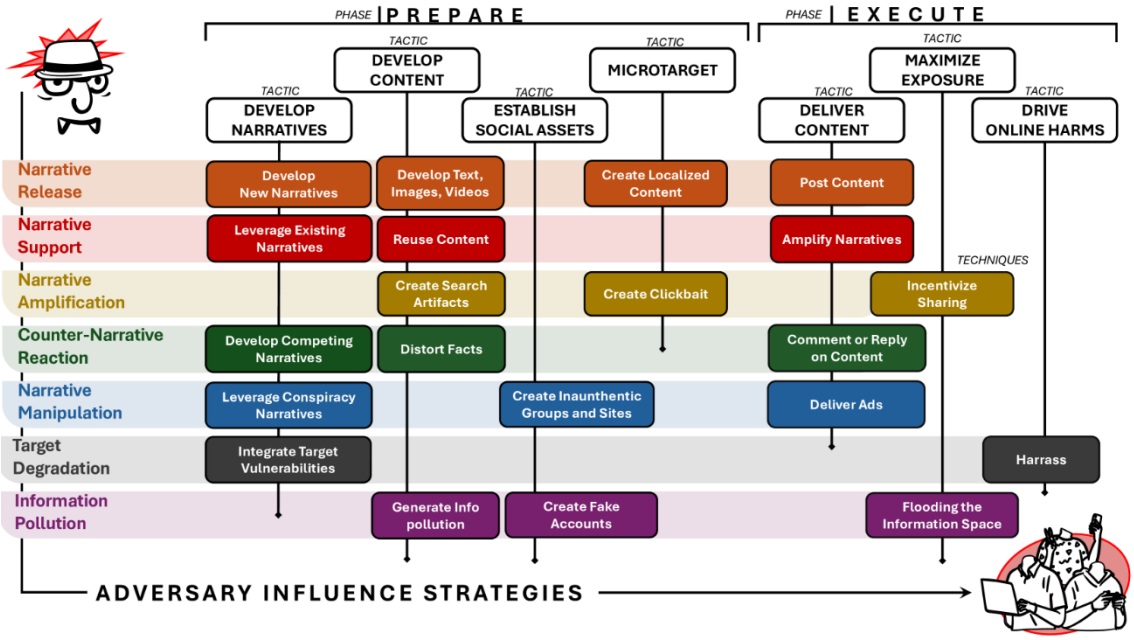


Figure 1. Seven strategies based on DISARM techniques for influence operations in social networks

The emergence of large language models has significantly expanded the capabilities of such campaigns. Matz et al. (2024) [10] demonstrate that generative AI systems can produce highly persuasive and personalized messages, outperforming generic communication strategies. This suggests that influence operations can now be scaled with minimal human intervention while maintaining high levels of contextual relevance.

Despite these advances, most research focuses either on content production or information spread, rarely addressing both simultaneously. This separation limits the ability to study influence campaigns as dynamic systems involving both generation and propagation.

2.2 Agent-Based Social Simulation

Agent-based modeling (ABM) has long been used to study complex social systems, where individual agents interact according to predefined rules [11]. Traditional ABMs, however, rely on simplified behavioral assumptions and lack the ability to generate realistic human-like communication [12].

Recent developments integrate large language models into agent-based simulations, enabling more natural and context-aware interactions. Systems such as MOSAIC (Liu et al., 2025) [13] and OASIS (Yang et al., 2024) represent significant progress in this direction. MOSAIC introduces LLM-driven agents with persona profiles that simulate

user behavior in social networks, allowing the study of content moderation and misinformation dynamics (Figure 2). OASIS extends this concept by scaling simulations to large populations, demonstrating emergent phenomena such as polarization and collective behavior (Figure 3).

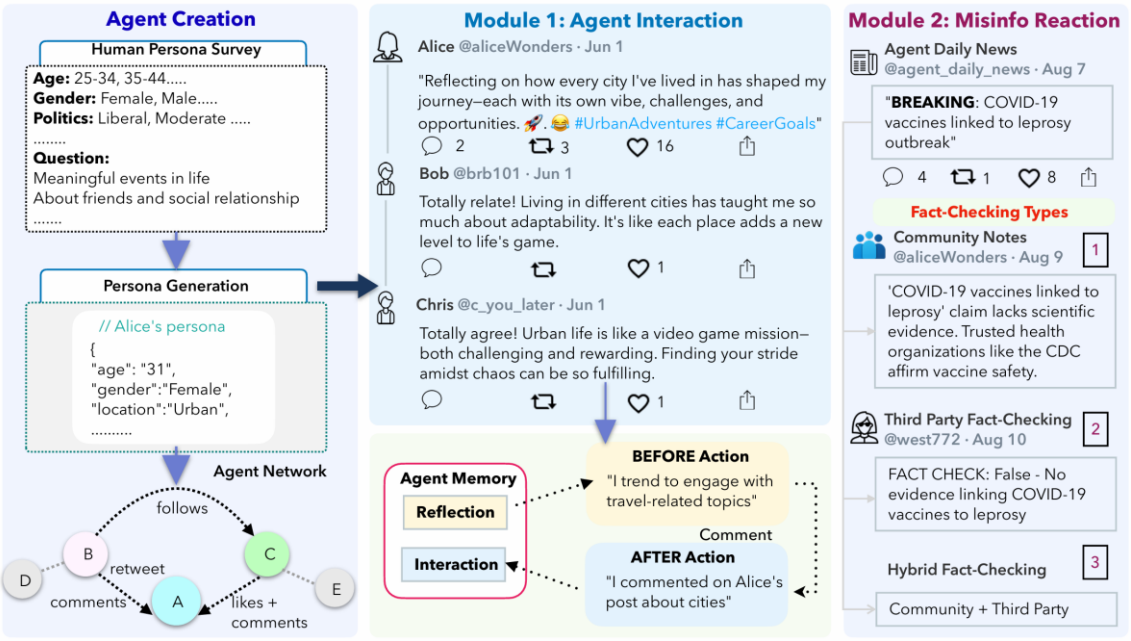


Figure 2. Structure of MOSAIC

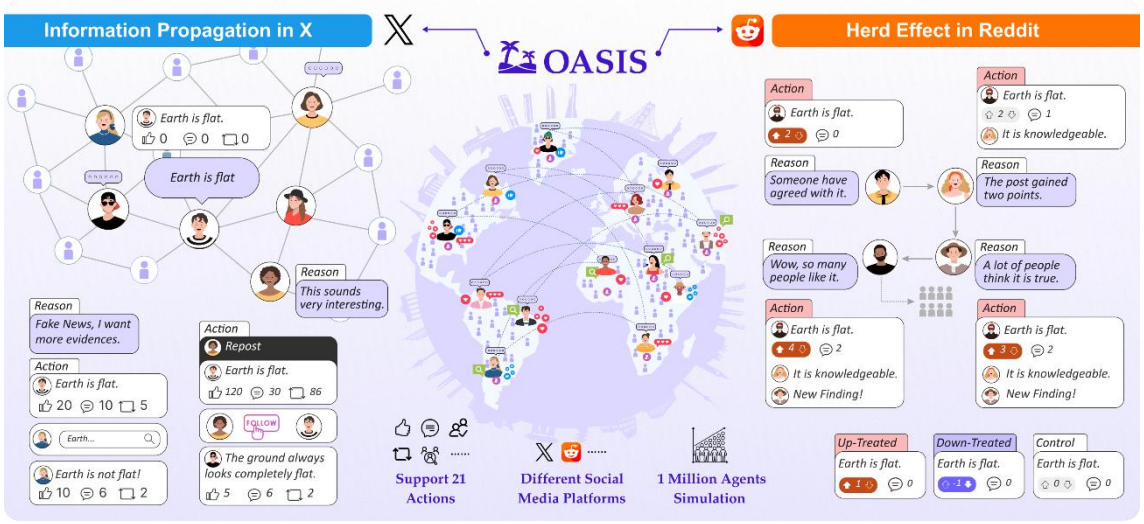


Figure 3. Structure of OASIS

Other studies explore coordination and competition among LLM agents. Orlando et al. (2025) [14] show that networked agents can exhibit emergent coordination patterns in influence scenarios, even without explicit communication protocols. Similarly, Qasmi et al. (2025) [15] model adversarial interactions between misinformation spreaders and debunking agents, highlighting the role of resource constraints and confirmation bias in shaping outcomes.

While these approaches significantly improve realism, they often lack fine-grained control over agent behavior. Most systems rely on static prompts or predefined roles, limiting their ability to simulate nuanced stylistic variation or community-specific adaptation.

2.3 Controllable Text Generation

Controllable text generation (CTG) focuses on guiding language model outputs to satisfy specific attributes such as sentiment, style, or stance. This is particularly relevant in the context of influence operations, where subtle variations in tone and framing can significantly affect persuasion outcomes.

Liang et al. (2024) [16] provides a comprehensive survey of CTG methods, categorizing them into four main approaches: fine-tuning, prompt conditioning, reinforcement learning, and decoding-time intervention. Fine-tuning methods, including parameter-efficient techniques such as LoRA (Hu et al., 2021) [17], enable strong and stable control by adapting model weights to target attributes. Prompt-based approaches offer flexibility but often lack reliability and consistency. Reinforcement learning methods optimize generation toward specific objectives but introduce challenges in reward design and training stability.

These techniques have demonstrated effectiveness in controlling isolated attributes of generated text. However, they are typically applied at the level of individual outputs and do not account for persistent agent behavior or interaction within a social environment. As a result, they are insufficient for modeling influence campaigns that require consistent stylistic identity and adaptation over time.

2.4 Trait and Style Modeling

Trait and style modeling focuses on the extraction and quantification of linguistic and psychological characteristics from text. These traits are essential for understanding user behavior in online environments and play a critical role in influence operations, where variations in tone, stance, and emotional framing can significantly affect persuasion.

A large body of work addresses the measurement of basic textual attributes, such as sentiment and subjectivity [18]. Classical approaches include lexicon-based methods like SentiWordNet and supervised models such as the Stanford Sentiment Treebank classifiers [19], [20]. More recent approaches rely on transformer-based models (e.g., BERT) for fine-tuned or zero-shot sentiment classification, enabling scalable analysis across large corpora [21], [22].

Beyond sentiment, stance detection and ideological classification have become important tasks in analyzing political and social discourse. These methods aim to determine whether a text expresses support, opposition, or neutrality toward a given topic. Benchmarks such as the SemEval stance detection tasks provide standardized datasets for evaluating such models, while more recent approaches leverage pretrained language models for improved generalization across domains [23].

Another key dimension is toxicity and harmful language detection, often operationalized through tools such as the Perspective API [24] or models trained on datasets derived from Jigsaw's toxic comment classification corpus [25]. These systems estimate attributes such as toxicity, insult, threat, and identity-based attacks. Such forms of hostile language are strongly linked to polarizing online discourse and have been studied in relation to misinformation and coordinated manipulation [26].

In addition to surface-level attributes, research also explores psychological and personality trait inference. Frameworks such as the Big Five (OCEAN) [27] (Figure 4) model have been widely used to infer personality characteristics from text, using features derived from linguistic analysis tools like LIWC [28] or learned representations from neural models. Such approaches enable the construction of richer user profiles, capturing not only what is said but how it is expressed.

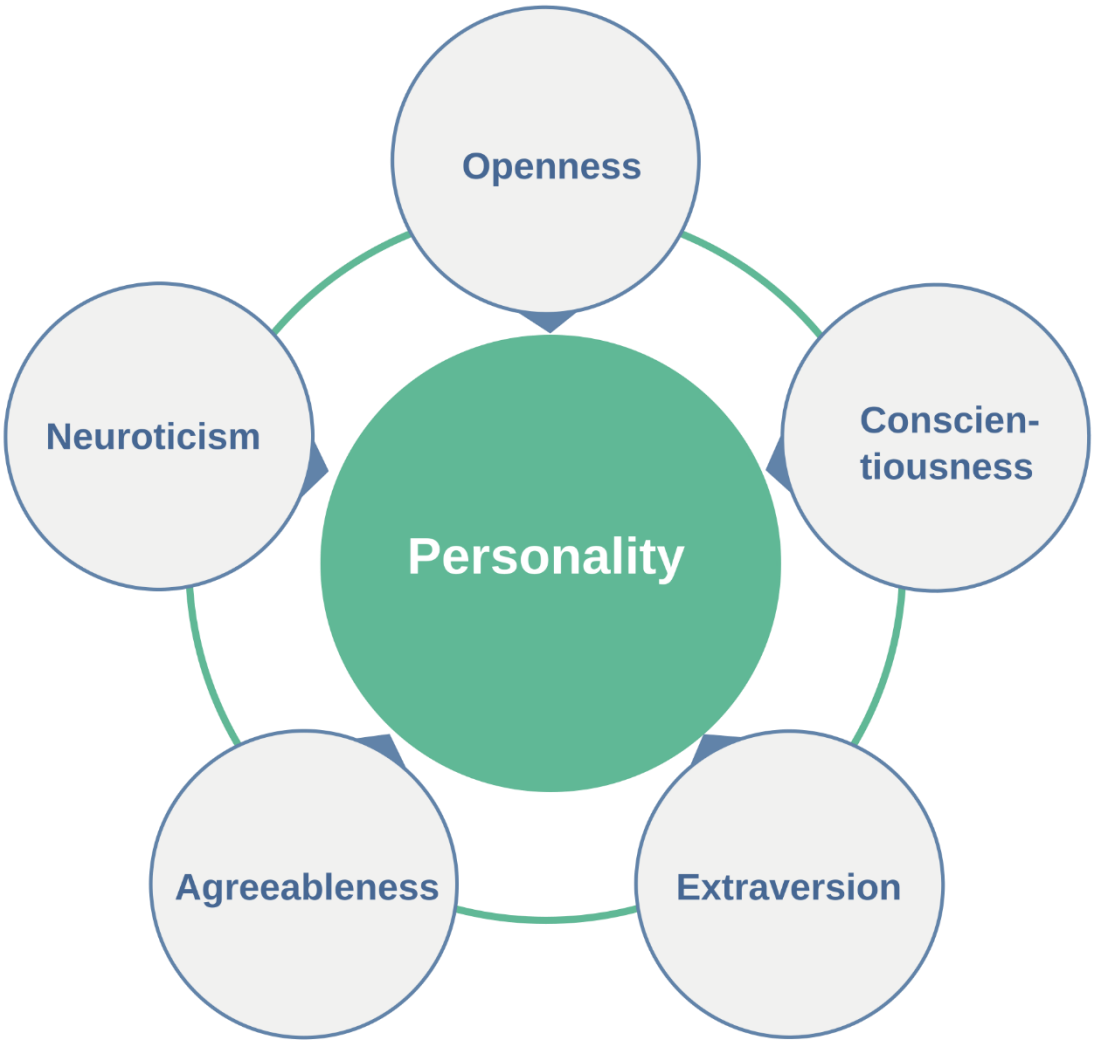


Figure 4. Big Five OCEAN

Recent advances increasingly rely on large language models for multi-dimensional trait extraction, combining zero-shot classification with fine-tuned predictors [29]. These methods allow simultaneous estimation of multiple attributes, including sentiment, emotional tone, stance, politeness, and rhetorical style, without requiring extensive labeled data for each trait.

Commonly modeled attributes include:

1. sentiment and emotional tone (e.g., joy, anger, fear),
2. stance and agreement,
3. toxicity and politeness,
4. subjectivity and certainty,
5. moral and ideological framing.

These traits [30], [31], [32] have been used extensively to profile users and communities, identify coordinated behavior, and analyze the dynamics of online discourse [33], [34].

However, most existing work treats trait modeling as a purely analytical task. While traits are extracted to describe content or users, they are rarely used as control variables in generative systems. This represents a key limitation in the context of influence operations [35].

In the domain of controllable text generation, it has been shown that manipulating stylistic attributes can significantly affect how messages are perceived and their persuasive impact (Liang et al., 2024). Yet, these approaches typically operate at the level of individual outputs and do not integrate trait signals into persistent agent behavior.

This thesis extends prior work by treating traits not only as descriptive features but as generative controls, embedding them into the training and behavior of agents. By doing so, it enables the simulation of influence strategies that rely on consistent stylistic adaptation across interactions and communities.

3 System Design and Pipeline

This chapter presents the overall architecture of the MATRIX system, including its design principles, data sources, and end-to-end processing pipeline. The system is designed as a staged research platform for simulating influence operations in online communities.

MATRIX is not a single model training procedure, but a multi-stage pipeline, where large-scale Reddit data are progressively transformed into simulation-ready artifacts. The architecture integrates data ingestion, community discovery, feature extraction, model training, and multi-agent simulation within a unified framework.

A central architectural decision is the separation between data storage and computation. Reddit archives are treated as immutable inputs, while all analytical logic is executed externally in Python. This design aligns with established large-scale data processing paradigms, where storage and computation are decoupled to improve flexibility and reproducibility [36].

Additionally, the system employs parameter-efficient model adaptation, specifically LoRA-based fine-tuning, which significantly reduces computational requirements compared to full fine-tuning.

3.1 High-Level Architecture

The MATRIX system is organized as a sequential processing pipeline composed of clearly defined stages. Each stage produces explicit outputs that serve as inputs to subsequent stages, ensuring modularity and reproducibility.

The system can be decomposed into the following layers:

1. raw data layer (Reddit archives),
2. processing and normalization layer,
3. operational storage layer (MongoDB),
4. feature and modeling layer (traits and judge models),
5. agent training layer,
6. simulation and evaluation layer.

The overall processing pipeline is illustrated in Figure 5.

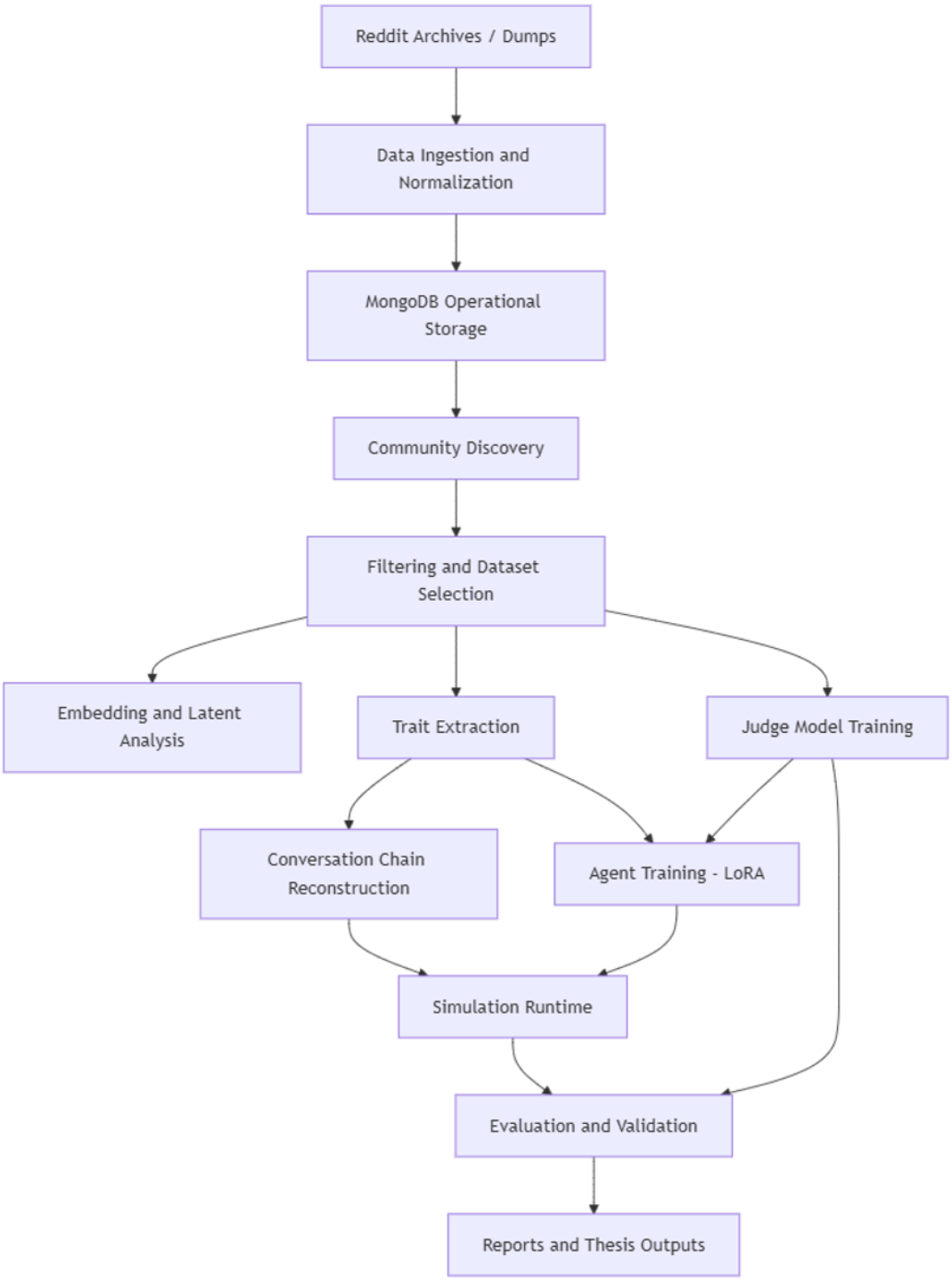


Figure 5. Processing pipeline

The pipeline begins with archived Reddit data, which are ingested and normalized into structured records. These records are stored in MongoDB [37] and subsequently processed through stages including community discovery, filtering, trait extraction, and model training. The outputs are combined in a simulation runtime, where agent interactions are generated and evaluated.

MongoDB is used as an operational storage layer, not as an analytical engine. Its role is limited to indexed retrieval and storage of intermediate results, while computationally intensive operations are performed in Python. This design reflects the requirement that complex transformations should not be implemented within the database engine.

Table 1. Storage and compute strategy

Approach	**Characteristics**	**Advantages**	**Limitations**	**Suitability**
MongoDB + Python	Document store with external compute	Flexible, supports experimentation	Less efficient for full scans	High
Columnar systems (Parquet)	Column-oriented analytics	Efficient for batch queries	Additional complexity	Moderate
Distributed frameworks (Spark)	Cluster-based processing	High scalability	Infrastructure overhead	Low
Database-centric logic	In-database computation	Centralized execution	Limited flexibility	Low

The selected approach prioritizes flexibility and iterative experimentation over maximum analytical throughput.

3.2 Data Sources

The primary data source for MATRIX consists of large-scale Reddit archives, including comments and submissions. Reddit data are widely used in computational social science due to their scale, structure, and accessibility [38].

The Pushshift dataset provides a comprehensive archive of Reddit content and has been extensively used in prior research. Additional datasets, such as Arctic Shift [39], provide updated versions with improved schema consistency.

Reddit offers several properties that are particularly relevant for this work:

1. explicit community segmentation via subreddits,
2. hierarchical discussion structure,
3. measurable interaction signals,
4. large-scale publicly available content.

From an architectural perspective, the data pipeline is divided into three tiers:

1. raw tier. Immutable archived datasets,
2. normalized tier. Structured records stored in MongoDB,
3. derived tier. Curated subsets and simulation-ready data.

MongoDB is used as an operational store, supporting indexed queries and intermediate storage. All computational tasks such as ranking, filtering, and feature extraction are performed externally in Python.

Live Reddit API data are not used as a primary data source. The Reddit API requires authentication and imposes usage constraints [40], and its Data API Terms restrict the use of API-derived content for machine learning purposes without explicit permission [41]. Therefore, archived datasets are preferred for reproducibility and consistency.

3.3 Design Principles

The architecture of MATRIX is guided by several key design principles.

3.3.1 Separation of Storage and Computation

The system follows a database-as-storage, Python-as-compute paradigm. MongoDB is used only for storage and indexing, while all analytical logic is implemented externally. This improves flexibility and avoids limitations of database-based computation.

3.3.2 Reproducibility

Each stage produces explicit outputs with defined configurations, enabling experiments to be reproduced and validated independently. Reproducibility is a fundamental requirement in machine learning systems [42].

3.3.3 Modularity

The system is structured as independent stages, each with well-defined inputs and outputs. This reduces coupling and simplifies development and validation.

3.3.4 Parameter-Efficient Adaptation

MATRIX uses LoRA-based fine-tuning instead of full model retraining.

Table 2. Model adaptation strategies

Method	**Trainable Parameters**	**Resource Requirements**	**Advantages**	**Limitations**
Full fine-tuning	High	Very high	Maximum flexibility	Not feasible on limited hardware
LoRA	Low	Low	Efficient adaptation	Requires adapter management
Adapter layers	Medium	Moderate	Modular design	Additional complexity
Prompt-based control	None	Minimal	No training required	Limited control

LoRA enables efficient adaptation of large language models while maintaining performance.

3.3.5 Data Usage and Legal Consideration

The system operates on publicly available Reddit archives obtained from third-party datasets commonly used in academic research. These datasets are used exclusively for non-commercial research purposes, and no attempt is made to identify or track individual users.

The design of MATRIX does not include explicit anonymization or pseudonymization mechanisms beyond those inherent in the source data. The system does not attempt to link identities across external datasets or infer real-world identities.

Live Reddit API data are not used for model training in scope of the MATRIX. This is consistent with Reddit's Data API Terms, which impose restrictions on the use of API-derived content for machine learning applications without explicit permission.

The system is designed as an offline research environment, and all simulations are conducted without interaction with live platforms. This ensures that generated content remains within a controlled experimental context.

This work follows standard academic practice in computational social science, where large-scale public datasets are analyzed to study aggregate patterns of behavior. Nevertheless, the limitations of such datasets, including potential presence of personal data, are acknowledged.

3.3.6 Offline Simulation

MATRIX is designed exclusively as an offline system for experimentation and analysis. It does not interact with live platforms or users. All generated outputs remain within the simulation environment.

4 Community Discovery and Dataset Preparation

This chapter describes the process by which the MATRIX system identifies relevant Reddit communities and prepares the study corpus for later modeling stages. Rather than treating subreddit selection as a manual or ad hoc step, MATRIX implements a staged pipeline in which large-scale archived Reddit data are progressively narrowed into a final, analysis-ready corpus. The chapter therefore covers not only the selection of communities, but also the operational logic used to move from raw monthly dumps to filtered, semantically validated, and model-ready subsets.

The empirical basis of this work is a three-month historical slice of Reddit comments and submissions covering May, June, and July 2025. The monthly bundles are distributed through Academic Torrents [43] and documented through the Arctic Shift project. The May 2025 release is distributed as a two-file bundle of 56.89 GB, the June 2025 release as a two-file bundle of 56.46 GB, and the July 2025 release as a two-file bundle of 58.35 GB, for a combined compressed source volume of approximately 171.7 GB [44], [45], [46]. Because this source volume is too large for direct manual inspection and inefficient for repeated end-to-end rescanning, community discovery is treated as a funnel consisting of ingestion, normalization, candidate generation, layered filtering, and final extraction.

The chapter also distinguishes between two operating modes of the prototype. The first is an exploratory fast path, based on a 14-day slice used for rapid iteration and debugging of selection logic. The second is the final extraction path, which operates on the selected communities across the broader May–July 2025 corpus. This distinction is methodologically legitimate: the short window accelerates development, whereas the three-month corpus provides the actual substrate for later modeling and simulation.

The overall workflow is shown in Figure 6.

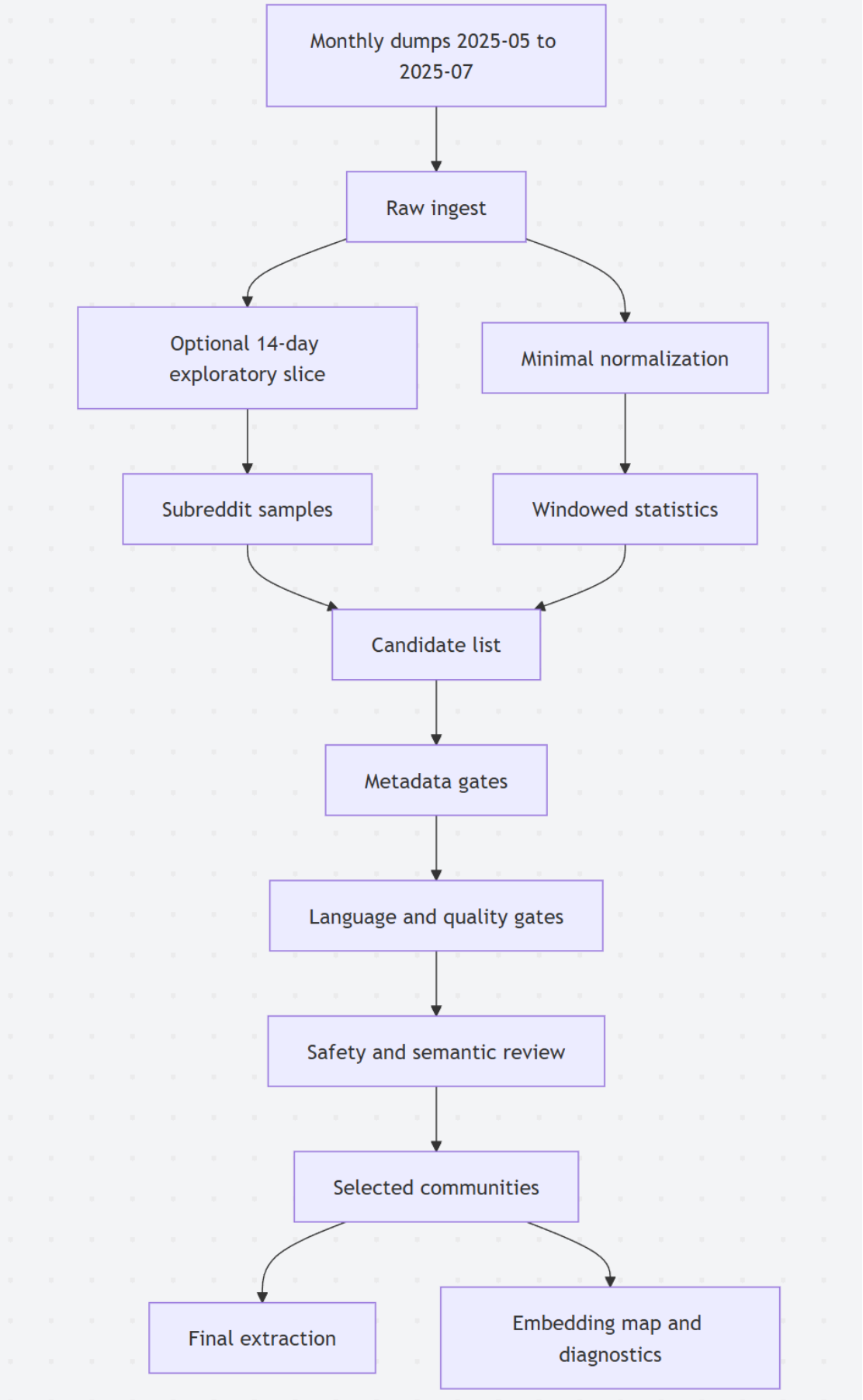


Figure 6. Data workflow

4.1 Subreddit Discovery

The purpose of subreddit discovery in MATRIX is not to enumerate all active Reddit communities, but to construct a candidate set that is broad enough to preserve topical diversity and narrow enough to remain computationally tractable. In this context, a useful community is one that exhibits sufficient content volume, temporal activity, and discourse structure to support later stages such as judge-model training, trait extraction, and agent adaptation.

This work uses archived Reddit dumps rather than live platform activity as the empirical universe. Monthly Reddit dumps have long served as research infrastructure for large-scale Reddit analysis, and the Pushshift dataset paper remains the canonical reference for describing Reddit monthly dumps and associated tooling as an enabler for academic work [47]. For MATRIX, the use of archived monthly bundles creates a fixed and reproducible historical window, avoiding the instability and throttling constraints associated with live collection.

The discovery strategy is best characterized as activity-first with optional semantic expansion. The core intuition is that communities must first meet basic statistical requirements before more expensive semantic analysis is justified. In practice, this means ranking communities based on observable activity signals such as document count, score variation, temporal spread, and the share of negatively scored content. This type of heuristic does not claim to measure "importance" in any universal sense; rather, it functions as a reproducible candidate-generation mechanism.

The current prototype already implements this logic in a lightweight form by scanning a normalized minimal collection in rolling windows and computing subreddit-level statistics. This choice is methodologically defensible because it is cheap, transparent, and suitable for workstation-scale experimentation. Activity-first pruning is also consistent with broader data-engineering practice: first reduce the candidate space with inexpensive statistics, then apply semantically richer methods to the retained subset.

A broader design space exists beyond this baseline. Candidate expansion could incorporate subreddit metadata, moderation rules, shared-author graphs, or semantic nearest-neighbor search over subreddit centroids. For graph-based grouping, the Louvain method provides a fast and widely used baseline for community detection in large

networks [48], while the Leiden algorithm is preferable when partition quality and connectedness become central, as it addresses the disconnected-community weakness of Louvain and typically produces better-connected partitions [49]. In the current thesis, however, such graph methods are treated as optional extensions rather than as the primary discovery mechanism.

A useful operational summary of discovery options is given in Table 3.

Table 3. Community discovery strategies

Method	**Advantages**	**Limitations**	**Role in MATRIX**
Activity-threshold scanning	Fast, reproducible, directly applicable to normalized corpus	May miss niche but relevant communities	Primary first-pass candidate generation
Keyword seed search	Transparent and easy to explain	Low recall, sensitive to vocabulary drift	Supplementary seed generation
Metadata/rules expansion	Interpretable, adds moderation and NSFW context	Depends on metadata availability and freshness	Auxiliary enrichment
Shared-author graph with Louvain	Captures behavioral adjacency, computationally efficient	May produce weakly connected communities	Exploratory grouping
Shared-author graph with Leiden	Better partition quality and connectivity than Louvain	More implementation complexity	Preferred graph option if graph partitioning becomes central

Embedding-based centroid retrieval	Strong semantic recall, supports "similar community" search	Sensitive to embedding quality and community breadth	Secondary semantic expansion

4.2 Community Filtering

Discovery alone is insufficient because many active communities are unsuitable for later modeling. A candidate subreddit may be unsuitable because it contains too little data, shows excessive duplication, is off-topic, is dominated by spam-like or automated content, or falls outside the intended analytical scope of the study. Community filtering in MATRIX is therefore designed as a layered process rather than a single binary decision.

The most defensible filtering strategy is a cascade of progressively more expensive checks. Early stages apply simple, transparent constraints such as minimum data thresholds, length requirements, literal deduplication, and obvious noise or bot heuristics. Later stages apply semantic review, coherence checks, and safety-oriented inspection. This layered structure reflects a basic engineering principle: expensive inference should only be applied after clearly unsuitable candidates have already been removed by cheaper tests.

At the subreddit level, the filtering process should consider at least the following dimensions:

1. minimum content volume and temporal spread,
2. language consistency,
3. topical coherence,
4. duplication and diversity,
5. safety and moderation state.

The current prototype already implements much of this logic. Exploratory subreddit sampling applies caps, literal deduplication, minimum-length thresholds, and simple noise handling. Later selection adds subreddit-level filtering through syntactic red flags,

semantic review, NSFW screening, diversity-ratio checks, and an embedding-based off-topic rate. Methodologically, this is one of the strongest parts of the current pipeline, because it shows that community selection is not a single classifier decision but a structured filtering cascade.

Metadata enrichment can also strengthen filtering. The Academic Torrents subreddit metadata, rules, and wikis release for January 2025 includes subreddit-level information such as descriptions, subscriber counts, NSFW (Not Safe For Work) flags, and related metadata. Such auxiliary metadata can be used as hard gates before semantic review, for example to exclude NSFW communities or to enrich discovery with descriptions and rules. This is preferable to rely exclusively on text classifiers when a transparent metadata signal already exists.

Table 4 presenting filtering comparison.

Table 4. Filtering and validation methods

Method	**Advantages**	**Limitations**	**Role in MATRIX**
Minimum data thresholds	Removes brittle communities	May exclude strategically interesting niche communities	Mandatory early gate
Literal deduplication	Cheap, removes boilerplate and spam artifacts	Misses paraphrased duplicates	Early cleaning stage
Bot/noise heuristics	Transparent and inexpensive	May generate false positives	Early rejection stage
Safety/NSFW review	Helps constrain study scope	Requires calibration	Review layer, not sole criterion
Embedding coherence/off-topic rate	Detects mixed or semantically	Sensitive to sample quality and embedding choice	Late-stage semantic validation

	unstable communities		

The output of this stage is a set of selected communities that are both analytically useful and operationally manageable.

4.3 Data Extraction

In MATRIX, data extraction is not merely a download step. It is a process of moving data between storage layers while progressively reducing complexity and preserving provenance. The most accurate description of the implementation is a three-layer data representation strategy:

1. Raw immutable layer- archived monthly Reddit dumps.
2. Normalized operational layer- reduced records in MongoDB.
3. Selected analytical layer- final corpora for downstream modeling.

This design is motivated by both scale and reproducibility. Raw monthly dumps provide the canonical source and should remain unchanged once ingested. However, repeated downstream processing over full raw objects is inefficient and introduces unnecessary schema friction. For this reason, the system constructs smaller operational representations that retain only the fields needed for later discovery, filtering, and modeling.

The current prototype already implements this logic. Raw content is streamed from compressed dumps into MongoDB, exploratory slices are built for rapid testing, and a minimal normalized representation is then used for later scanning and selection. Downstream extraction copies document only from retained communities into a final collection. This is best described as a minimal normalized representation strategy: keep the raw corpus intact, then derive smaller representations tailored to later tasks.

A representative minimal schema example is presented in Figure 7.

```
_id: "t3_1l0bc9j"
parent_id : null
root_id : "t3_1l0bc9j"
author : "MuertePorMexy"
subreddit : "EDH"
created_utc : 2025-06-01T03:00:00.000+00:00
score : 1
heading : "Edgar Markov Deck Help"
text : "I recently decided to build a Markov deck and I'm looking for some adv…"
```


Figure 7. Sample of minimum schema document

Such a schema is sufficient for later discovery, chain reconstruction, and sampling, while avoiding repeated rescanning of the full raw payload.

Archived monthly dumps offer a fixed historical snapshot and support reproducibility. By contrast, live Reddit API collection is subject to authentication requirements, listing-based pagination, rate limits, and policy constraints.

Nonetheless the whole pipeline extracts useful documents in number of approximately 246 million from raw 1.1 billion of documents.

A source-and-layer inventory is shown in Table 5.

Table 5. Data sources and representation layers

Layer	**Main content**	**Role in pipeline**	**Notes**
Raw archive	Monthly Reddit comments and submissions dumps	Canonical input substrate	Immutable and compressed
Normalized operational store	Reduced records in MongoDB	Discovery, sampling, filtering, chain preparation	Supports indexed retrieval
Selected analytical corpus	Documents from retained communities	Final input for later modelling stages	Constrained to relevant subreddits
Auxiliary metadata	Subreddit metadata	Optional moderation and topical enrichment	Used as supplementary signals

4.4 Embedding and Visualization

Embedding and visualization are used in MATRIX as tools for analysis, validation, and discovery support, not as substitutes for substantive selection criteria. Their purpose is to help verify that selected communities are neither trivially redundant nor obviously off topic before downstream training begins.

The general idea is straightforward: each candidate subreddit is represented by a sample of texts, these texts are mapped into an embedding space, and the resulting vectors are aggregated into a compact subreddit-level representation. The prototype uses transformer-based sentence embeddings for this purpose. Sentence-BERT is a natural reference point here because it was explicitly designed to make semantic similarity, clustering, and retrieval practical by generating comparable sentence vectors [50]. E5-style embedding models provide a stronger modern alternative for retrieval, clustering, and classification, and are well suited to semantic centroid search and nearest-neighbor expansion [51].

Once subreddit-level vectors are constructed, they can be projected into two dimensions for exploratory inspection. UMAP is an appropriate default for this task because it is computationally efficient and designed for dimensionality reduction while preserving meaningful structure in the data [52]. In smaller diagnostic subsets, t-SNE can still be useful when local neighborhood fidelity is the main concern [53], but UMAP is more practical as the default visualization method in a pipeline of this size. The example of UMAP visualization is presented on Figure 8.

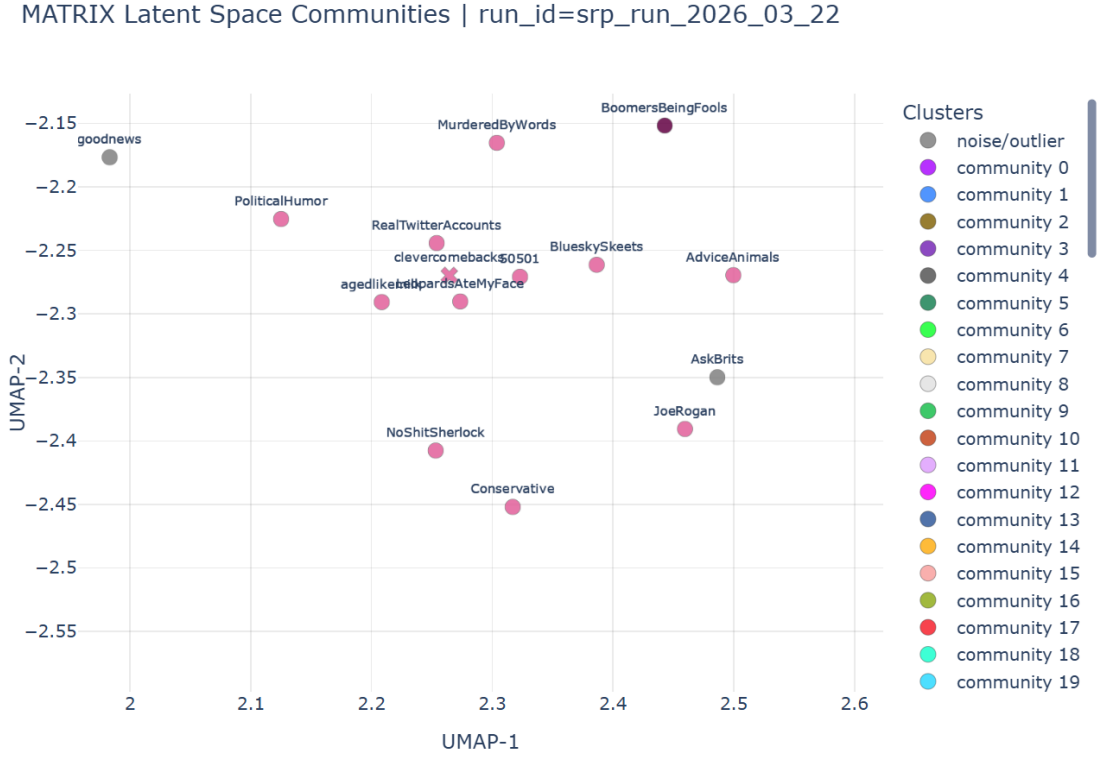


Figure 8. UMAP visualization

If unsupervised structure becomes more important in future iterations, additional methods are available. HDBSCAN is a reasonable choice when density varies, and noise points should be identified explicitly [54]. BERTopic is useful when clustering must also be paired with interpretable topic labels, because it combines embeddings, clustering, and class-based TF–IDF representations into one exploratory workflow [55].

5 Judge Models for Community Context

Judge models in MATRIX are community-conditioned evaluators designed to estimate how well a candidate post or comment aligns with the discourse norms, stylistic expectations, and likely reception of a target subreddit. Within the scope of the prototype development, unlike generative models, which produce text, judge models operate as evaluative components that approximate how content would be perceived within a specific community context. Within the overall MATRIX architecture, they form the interface between generation and simulation: agents produce candidate outputs, while judges estimate their plausibility, contextual fit, and relative competitiveness.

This formulation is grounded in the paradigm of preference learning and reward modeling, where human or proxy preferences are learned as predictive signals rather than explicitly encoded objectives [56], [57]. Large-scale systems such as InstructGPT [58] demonstrate that learned evaluators can effectively guide model behavior. However, recent work on LLM-based evaluators shows that such models are subject to systematic biases, including verbosity bias, positional bias, and sensitivity to prompt structure [59]. These limitations highlight the importance of carefully constraining the role of judge models.

In the Reddit domain, the evaluation problem is further complicated by community heterogeneity. Prior research shows that different online communities encode distinct norms, moderation practices, and expectations of acceptable discourse [60], [61]. More recent studies on community-personalized preference learning demonstrate that different communities may prefer different answers to identical prompts [62]. In addition, Reddit-specific research shows that engagement depends on a combination of linguistic content, temporal dynamics, and thread structure [63], [64].

For these reasons, judge models in MATRIX are explicitly designed as community-local evaluators. They do not attempt to estimate universal quality or truthfulness but rather approximate the likelihood that a piece of content would be accepted within a specific subreddit. Engagement signals such as upvotes and replies are treated as proxies for community endorsement rather than indicators of factual correctness, consistent with prior findings that online engagement often correlates with emotional and narrative alignment rather than truthfulness [65].

Formally, a judge model implements a mapping:

$$(subreddit, context, candidate) \rightarrow \{score, probability, preference, uncertainty\}$$

where each output corresponds to a different operational perspective on community reception. These outputs are complementary rather than interchangeable and form the basis of downstream evaluation and reranking.

5.1 Feature Engineering

The judge subsystem relies on a multi-modal feature space combining semantic, lexical, rhetorical, and contextual signals.

Semantic embeddings are used to represent candidate texts and context, enabling similarity-based comparisons. This approach is supported by prior work on sentence embeddings for ranking and retrieval tasks. Lexical features, derived from TF–IDF representations, capture community-specific phrasing patterns that may not be fully encoded in dense embeddings.

Rhetorical features capture stylistic aspects such as politeness, certainty, and question structure. These features are motivated by research demonstrating the impact of discourse style on perceived quality [66].

Contextual features encode thread position, timing, and alignment with root content. These signals are critical because engagement on Reddit depends strongly on conversational structure and timing [67].

The integration of these feature families results in a hybrid representation that captures both semantic meaning and interaction dynamics.

Table 6. Feature families in judge models

Feature Family	**Description**	**Role**
Semantic	Sentence embeddings	Meaning and similarity
Lexical	TF-IDF	Community phrasing

Rhetorical	Style indicators	Discourse patterns
Contextual	Threads and time features	Interaction modeling
Structural	Post/comment flags	Lightweight signals

5.2 Model Training

Judge models are trained using a multi-objective framework combining regression, classification, and ranking.

The regression objective predicts a transformed engagement score:

$$y = sign(score) \cdot log(1+| score |)$$

This transformation reduces variance and stabilizes training for heavy-tailed distributions [68].

Binary classification predicts whether a candidate belongs to a high-performing percentile, improving robustness over raw score prediction [69].

The central objective is pairwise ranking, where the model learns to select the more community-appropriate candidate among semantically similar alternatives.

Table 7. Judge model objectives

Objective	**Target**	**Purpose**
Regression	Transformed score	Continuous estimation
Classification	Top percentiles	Filtering
Ranking	Pairwise preference	Reranking
Hybrid	Combined outputs	Robust scoring

5.3 Validation of Judges

Evaluation of the judge subsystem reveals distinct performance characteristics across tasks.

Classification performance shows moderate discrimination ability. However, ranking-based evaluation significantly outperforms both regression and classification.

The distribution of pairwise AUC across subreddits is shown in Figure 9, highlighting variability between communities and confirming that performance is context-dependent.

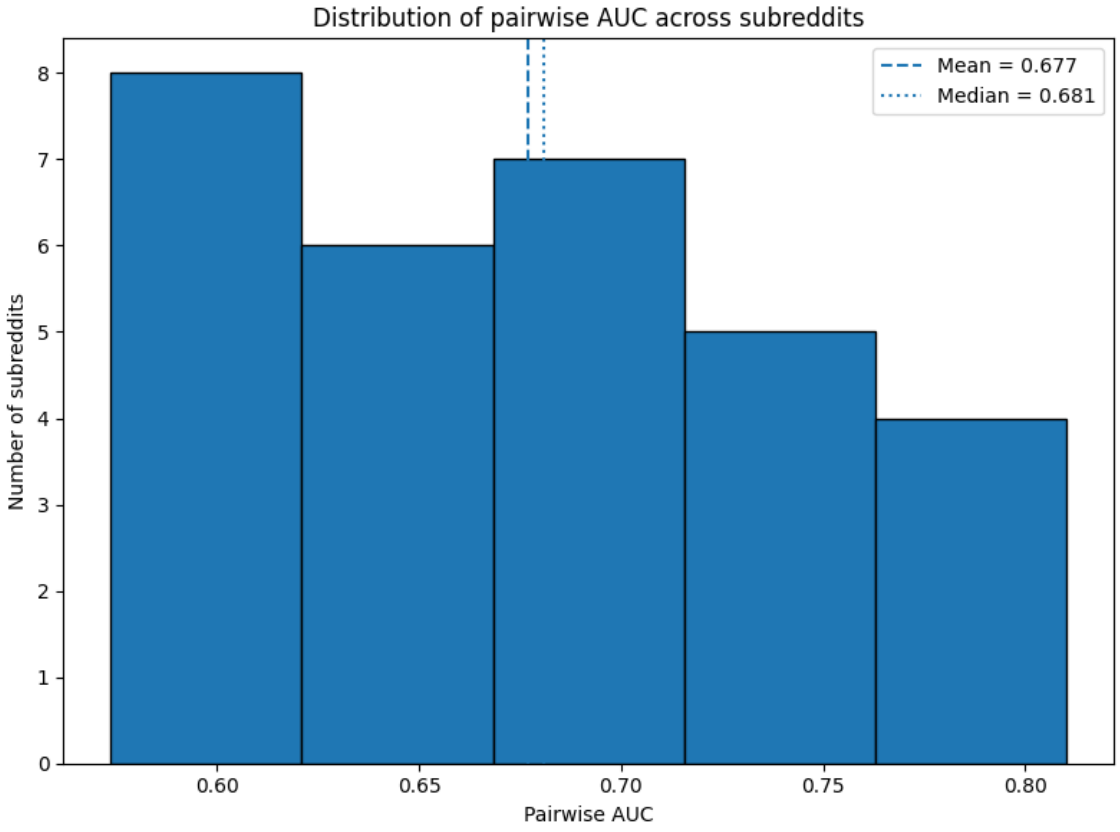


Figure 9. Distribution of pairwise AUC across subreddits

A direct comparison between regression and ranking performance is shown in Figure 10, which plots Spearman correlation against pairwise AUC (Area Under the Curve). The figure illustrates that ranking performance remains strong even when regression performance is weak.

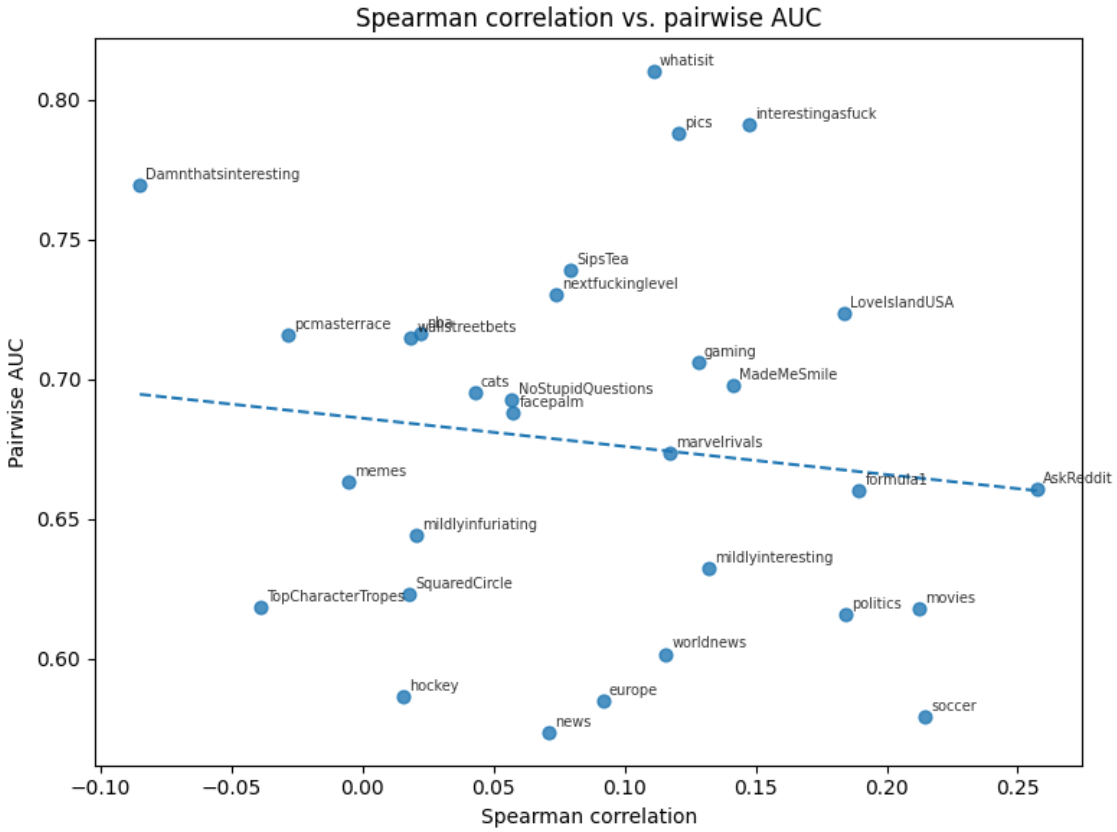


Figure 10. Scatter plot of Spearman correlation vs. pairwise AUC across subreddits

Grouped evaluation further demonstrates the effectiveness of the judge subsystem. As shown in Figure 11, top-1 selection accuracy exceeds random baseline performance, confirming that the model provides meaningful improvements in candidate selection.

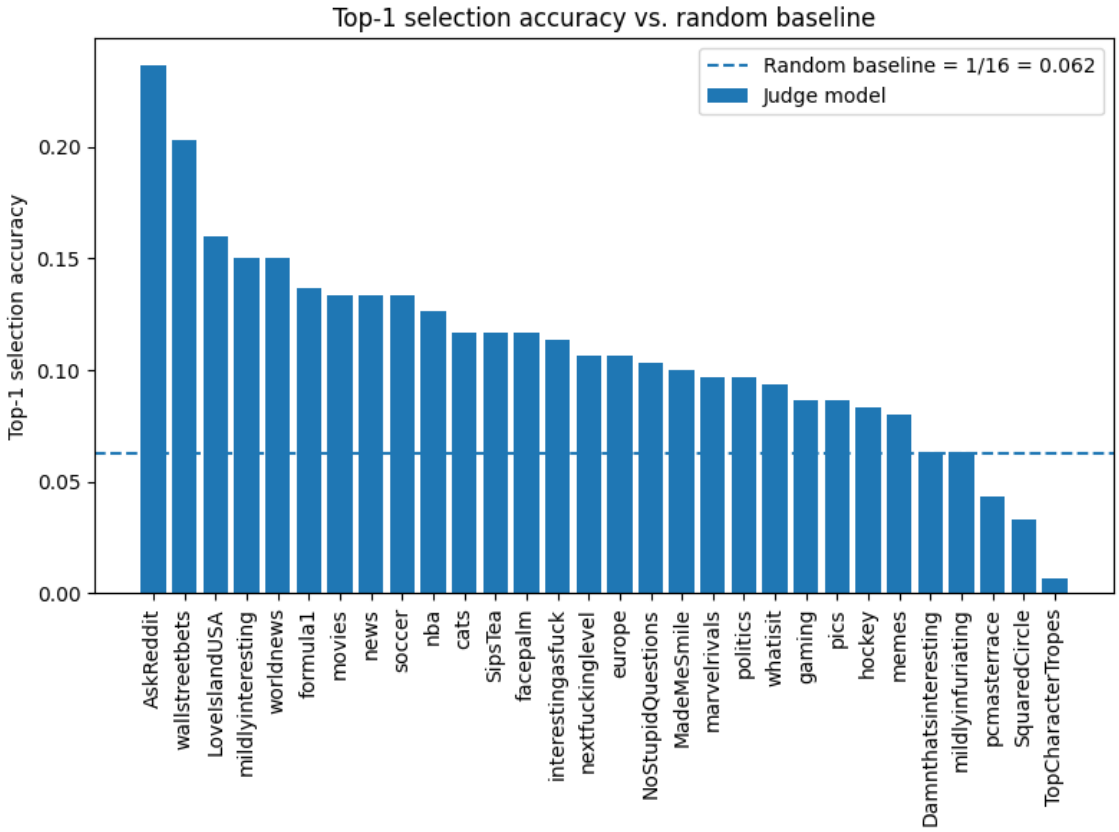


Figure 11. Bar chart comparing top-1 selection accuracy of judge model vs. random baseline

Table 8. Aggregate evaluation results

Metric	Mean	Interpretation
RMSE	~1.38	Stable but noisy
Spearman	~0.09	Weak ranking
AUC (classification)	~0.57	Moderate
Pairwise AUC	~0.68	Strong
Top-1 accuracy	~0.11	Above random
Top-3 overlap	~0.23	Effective reranking

The evaluation results support three key conclusions:

1. Absolute engagement prediction is inherently noisy.

2. Relative ranking is a more reliable evaluation signal.
3. Community-specific variation is significant and must be preserved.

These findings are consistent with prior work on preference learning and evaluator models.

6 Multi-Agent Training

This chapter details the training of large language model-based agents using parameter-efficient fine-tuning techniques. The goal of this stage is not to train a single monolithic generator, but to construct a bank of community-specialized agents that inherit a shared semantic prior from a common backbone while learning subreddit-local discourse style, conversational rhythm, and response conventions. In the MATRIX architecture, each selected subreddit is therefore treated as a distinct adaptation domain.

This design follows from the broader objectives of the thesis. MATRIX aims to simulate influence behavior inside online communities rather than to optimize generic chatbot helpfulness. A shared base model provides global fluency and broad world knowledge, while community-specific fine-tuning allows each agent to internalize locally plausible language use. This community specialization is also consistent with the recent direction of multi-agent social simulation research, where language-driven agents are used to model realistic social interaction rather than isolated text generation (MOSAIC, OASIS).

The current implementation realizes this design through parameter-efficient supervised fine-tuning of a shared backbone on subreddit-specific data. In the first training round, each agent is adapted from the same base model using 4-bit LoRA-style fine-tuning. In the second round, a gentler continued training pass is applied by initializing from the first-round adapter and lowering the learning rate. The result is a modular agent bank in which each subreddit-specific agent can later be evaluated separately, reused in conversation chains, or composed into larger simulations.

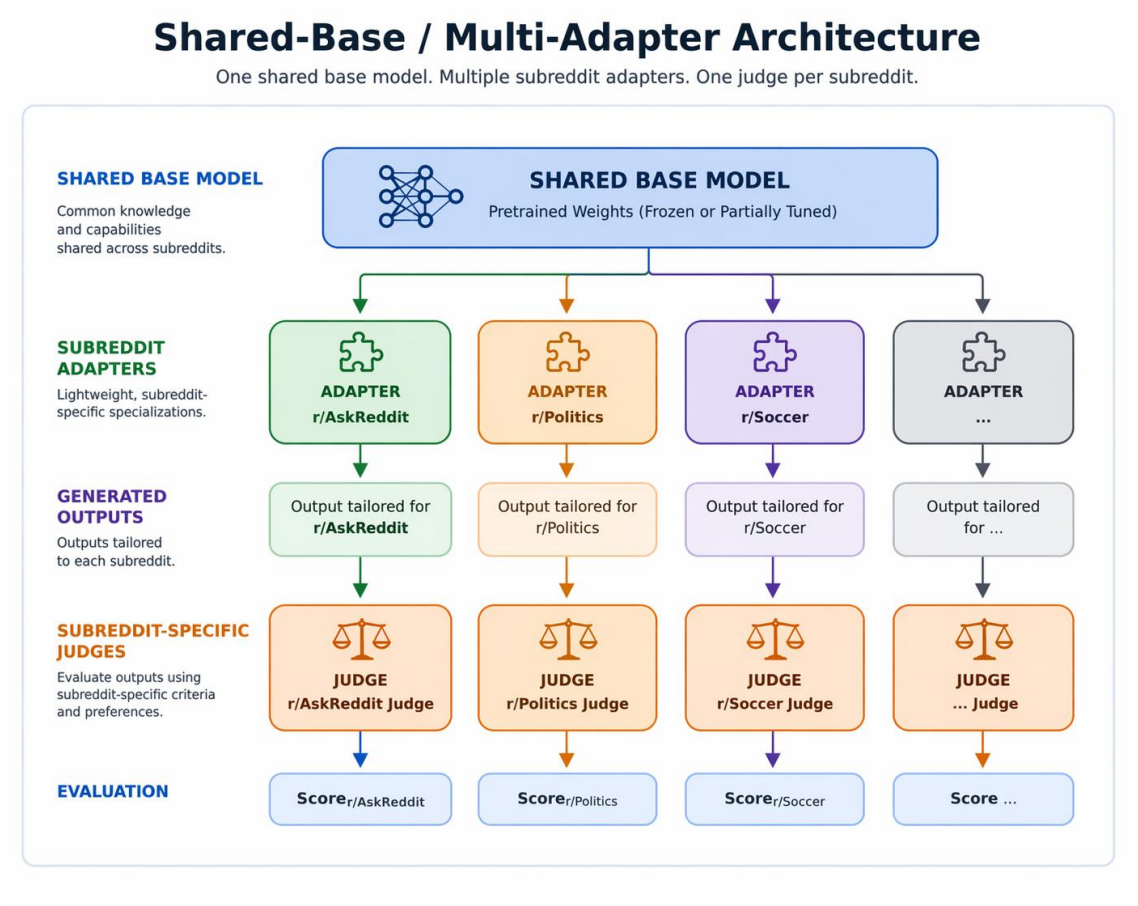


Figure 12. Multi-Adapter architecture

6.1 Base LLM and Adaptation

Base-model selection in MATRIX is not only a benchmarking decision but also a systems decision. The backbone must satisfy several constraints simultaneously. First, it must fit into single-GPU PEFT training on a workstation-class machine. Second, it must support instruction-following and chat-style prompting, because the training tasks in MATRIX are formulated as situated post and reply generation rather than plain next-token continuation. Third, it must remain strong enough that community specialization is meaningful, meaning that fine-tuning should refine discourse behavior rather than relearning basic language competence.

The implemented backbone in the current prototype is Qwen/Qwen2.5-3B-Instruct [70]. This model was selected as a practical balance between generation capability and computational efficiency: the Qwen2.5 family provides strong instruction-following and long-form generation performance, while the 3B instruction-tuned variant remains compact enough for local fine-tuning and repeated experimental use within the available

hardware budget [71]. According to the official model card, this model has 3.09B parameters, 36 layers, grouped-query attention, and a 32,768-token context window. The related 7B instruct model has 7.61B parameters and a 131,072-token context window. Official Qwen sources also note an important licensing distinction inside the Qwen2.5 family: the 3B and 72B variants are governed by Qwen's research-oriented licensing terms, while most other open-weight sizes in the family, including 7B, are released under Apache 2.0. By contrast, Mistral 7B [72] is explicitly released under Apache 2.0 and is presented by the vendor as a generally reusable open model.

These considerations make the choice of Qwen2.5-3B-Instruct a pragmatic academic prototype decision rather than an obviously dominant final choice. It is small enough to fit comfortably into 4-bit PEFT training on a 24 GB GPU, and it already supports the chat-style interaction pattern needed by the pipeline. At the same time, the licensing situation is less permissive than for alternatives such as Qwen2.5-7B-Instruct or Mistral-7B-Instruct-v0.3. For a thesis prototype this is acceptable; for broader redistribution or production use, a more permissively licensed 7B-class backbone would be the cleaner long-term choice. The chapter therefore treats Qwen2.5-3B-Instruct as the implemented baseline and Qwen2.5-7B-Instruct or Mistral-7B-Instruct-v0.3 as the most natural upgrade paths.

Table 9. Candidate base models for MATRIX agents

Base model	**Scale and context**	**License / access model**
Qwen2.5-3B-Instruct	3.09B, 32K context	Qwen research-oriented license
Qwen2.5-7B-Instruct	7.61B, 128K context	Apache 2.0
Mistral-7B-Instruct-v0.3	7B-class instruct model	Apache 2.0
Llama-3.1-8B-Instruct	8B, long-context instruct model	Gated / custom community terms
Gemma-class instruct model	Mid-size instruct family	Vendor terms acceptance required

The adaptation strategy is equally important. Full fine-tuning would require far more memory, storage, and experiment time than is reasonable for a workstation-scale prototype. MATRIX therefore adopts a PEFT-first design. LoRA is especially suitable because it inserts low-rank trainable updates into selected projection matrices, drastically reducing the number of trainable parameters compared to dense fine-tuning. QLoRA extends this idea by training LoRA adapters on top of a frozen 4-bit quantized backbone, making it possible to fine-tune comparatively large models with far smaller memory budgets while preserving strong task performance [73]. The current implementation uses exactly this practical pattern: 4-bit NF4 quantization with double quantization and LoRA applied to attention and MLP projection modules.

Other PEFT families were possible but were not chosen as the primary method. Prompt tuning and prefix tuning learn soft prompts or virtual prefixes rather than low-rank weight updates [74], [75]. Adapters add trainable modules between layers [76], while IA3 scales internal activations rather than learning full low-rank matrices [77]. These methods all offer different trade-offs, but for subreddit-specific adaptation the main requirement is persistent behavioral specialization under modest hardware constraints. LoRA and QLoRA provide the strongest quality-to-compute trade-off for that purpose.

Table 10. PEFT methods

Method	**Core idea**	**Advantages**	**Weaknesses**
Prompt tuning	Learn soft prompt vectors only	Minimal trainable footprint	Weaker persistent style control on smaller backbones
Prefix tuning	Learn virtual prefix tokens	Stronger than prompt tuning in some low-data settings	Less convenient for chat-style pipelines
Adapters	Insert trainable modules between layers	Strong modularity, reusable task blocks	Adds architectural complexity and inference overhead

IA3	Scale internal activations	Parameter-efficient	Less expressive for rich style transfer
LoRA	Learn low-rank updates on selected modules	Strong empirical adoption, compact adaptation	Requires target-module tuning
QLoRA	LoRA on frozen 4-bit quantized weights	Best quality-to-memory trade-off on single GPU	More brittle engineering stack

The implemented round-one configuration applies LoRA to q_proj, k_proj, v_proj, o_proj, up_proj, down_proj, and gate_proj, with rank 16, alpha 32, dropout 0.05, a learning rate of $1x10^{-4}$, one training epoch, and a maximum sequence length of 512 tokens. The model is loaded in 4-bit NF4 mode with double quantization enabled. The second round keeps the same data pipeline and maximum sequence length but lowers the learning rate to $5x10^{-5}$ and initializes from the first-round adapter rather than from the raw base model. This makes the second stage a conservative refinement pass rather than a distinct training method (Figure 13).

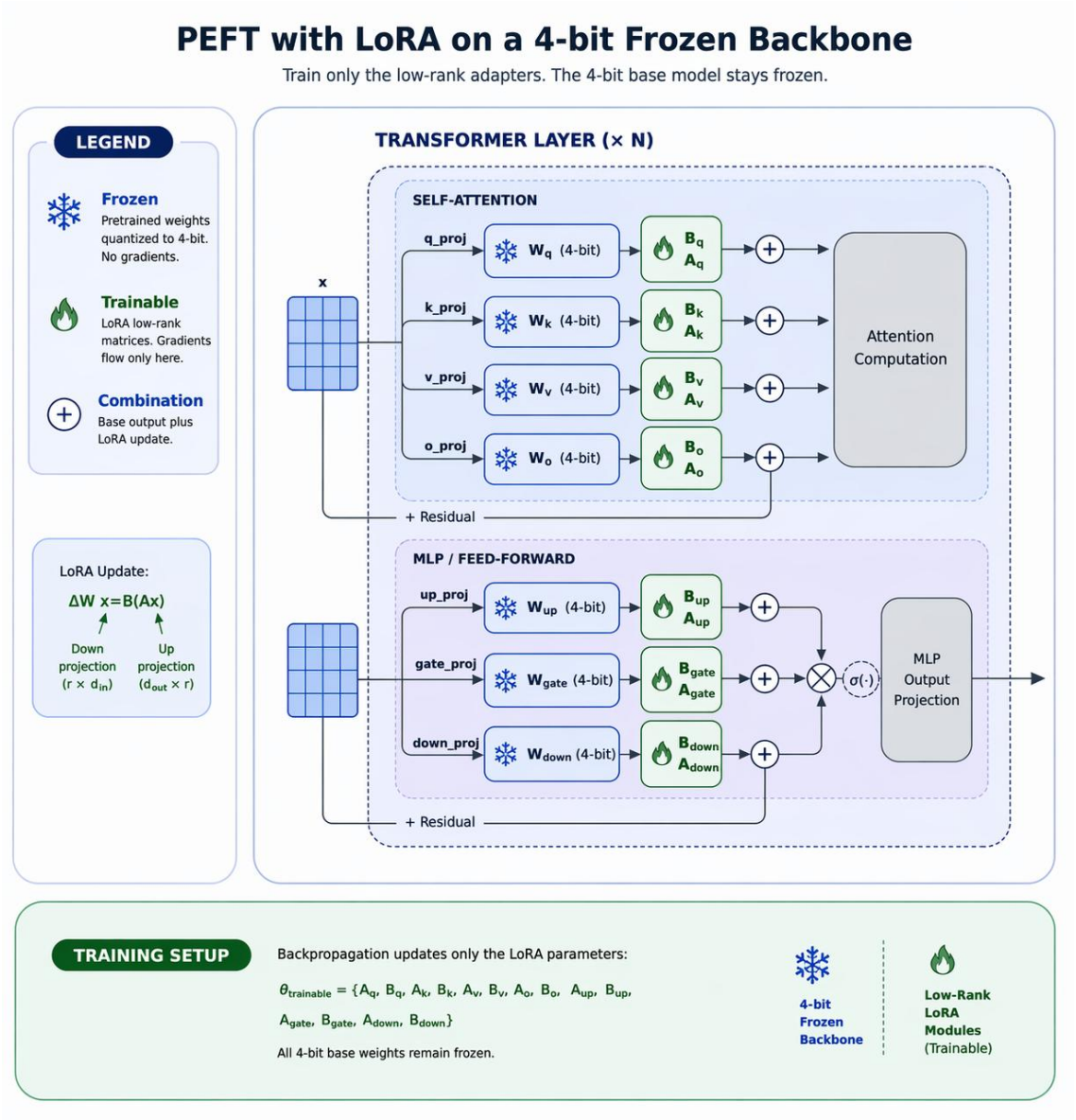


Figure 13. PEFT adaptation schematic

6.2 Training Agents

The training stage in MATRIX is organized around the idea that each subreddit is a separate adaptation domain. The prototype therefore trains one agent per selected subreddit, with a shared backbone and subreddit-specific LoRA weights. The training scripts cap the number of subreddits at 30, query data from the selected_things collection, and then perform curation and tokenization independently for each community. The first-round training script sets the core limits to 10,000 rows per subreddit, fetched from a larger 20,000-document pool, with an 80/20 bias toward high-score items and a smaller recency component. The curated set is then capped at 4,000 examples per subreddit, with

hard minimums of 1,000 raw documents and 500 curated examples. An evaluation split of 10% is applied after curation.

This data-selection policy is methodologically important. Social-media training corpora creates a tension between salience and freshness. If training draws only from high-score content, the resulting agent risks becoming a museum of historically successful community language. If it draws only from recent content, the model may underrepresent what the community rewards. The implemented design balances these two concerns by using a larger fetch pool, sorting primarily by score, and still reserving part of the pool for recent examples. This is a practical way to bias training toward community-endorsed discourse without discarding temporal variation entirely.

The current implementation reconstructs two basic task types: write_reply and write_post. If parent or root context is available, the example is framed as a reply-generation task. Otherwise, it is framed as a post-generation task. In the reply case, the prompt includes the subreddit identifier, task tag, root text if present, and parent text if present, followed by an instruction to write one plausible subreddit-native reply. In the post case, the prompt includes the subreddit, the task tag, an optional heading, and optional post-body context. This task framing is more appropriate than plain domain-adaptive next-token training, because it aligns the supervision format with the intended use of the agents later in the pipeline. The agent is not trained merely to continue generic Reddit text, it is trained to produce situated content inside a conversational frame.

The supervision strategy is equally deliberate. After the chat template is rendered, only assistant-side tokens are supervised. Prompt-side tokens are masked with -100 labels, and examples with fewer than 16 supervised assistant tokens are discarded. This prevents the training objective from overfitting to prompt reconstruction rather than response generation. It also removes trivial, empty, or low-information targets. Duplicate targets after normalization are removed during curation, and very short examples are discarded. As a result, the final dataset is optimized for response imitation rather than raw corpus compression.

Given the effective batch size of 2x4=8, the optimizer-step budget is moderate rather than excessive. At the high end, with roughly 3,600 training examples after a 10% evaluation split from a 4,000-example cap, the total number of optimizer steps is approximately

⌈3600/8⌉≈450. At the minimum useful threshold of 500 curated examples, the number of steps is approximately ⌈450/8⌉≈57. This makes the training budget realistic on a single 24 GB GPU.

The training workflow itself is also modular. The first script trains the initial LoRA adapters. The first evaluation script compares these adapted agents against the untuned base model. The second training script performs the gentler continuation pass. And the second evaluation script performs a stronger tournament-style comparison. The intended execution order is therefore:

1. first-round training,
2. base-versus-agent evaluation,
3. second-round training,
4. tournament-style evaluation,
5. optional merge or deployment preparation.

This staged design is useful for experimentation because it allows the effects of the second pass to be measured directly rather than folded into a single opaque result.

In the one-generation comparison, which evaluates 100 prompts per subreddit over 10 subreddits, the mean judge-score delta of LoRA over the base model is approximately +0.0404, with an average win rate of 0.56. The strongest positive deltas appear in communities such as politics, soccer, wallstreetbets, and NoStupidQuestions, while NBA is slightly negative and mildlyinfuriating is near neutral.

The tournament setting is stronger because it generates five samples from the base model and five from the adapted model for each prompt, scores all ten with the community judge, and then computes all 25 cross-pairs. In this round-one tournament evaluation, the mean cross-win rate is approximately 0.5609, with a mean score delta of +0.0387. After the second-round continuation pass, the mean cross-win rate increases slightly to 0.5650, with a mean score delta of +0.0379. The strongest improvements again appear in politics, soccer, wallstreetbets, and NoStupidQuestions. NBA remains negative, and worldnews weakens after round two. This pattern suggests that community specialization is working, but not uniformly, and that additional training is not automatically beneficial for every subreddit.

Table 11. Preliminary evaluation

Evaluation setting	**Communities**	**Mean delta**	**Mean win / cross-win rate**	**Interpretation**
Single-generation base vs LoRA	10	+0.0404	0.5600	LoRA agents outperform base on average
Tournament, round one	10	+0.0387	0.5609	Pairwise advantage remains after multi-sample comparison
Tournament, round two	10	+0.0379	0.5650	Second round gives small average gain, but unevenly

The interpretation of these results should remain bound. They are strong enough to justify the architectural choice of subreddit-specific adaptation, but not strong enough to claim universal success. Several failure modes are already visible. Some subreddits benefit much more than others. Some communities appear to over-specialize after the second pass. In some cases, the adapted model becomes more community-native according to the judge but also more narratively idiosyncratic or autobiographical, which may not always be desirable. Results of evaluation are presented in Figure 14 and Figure 15.

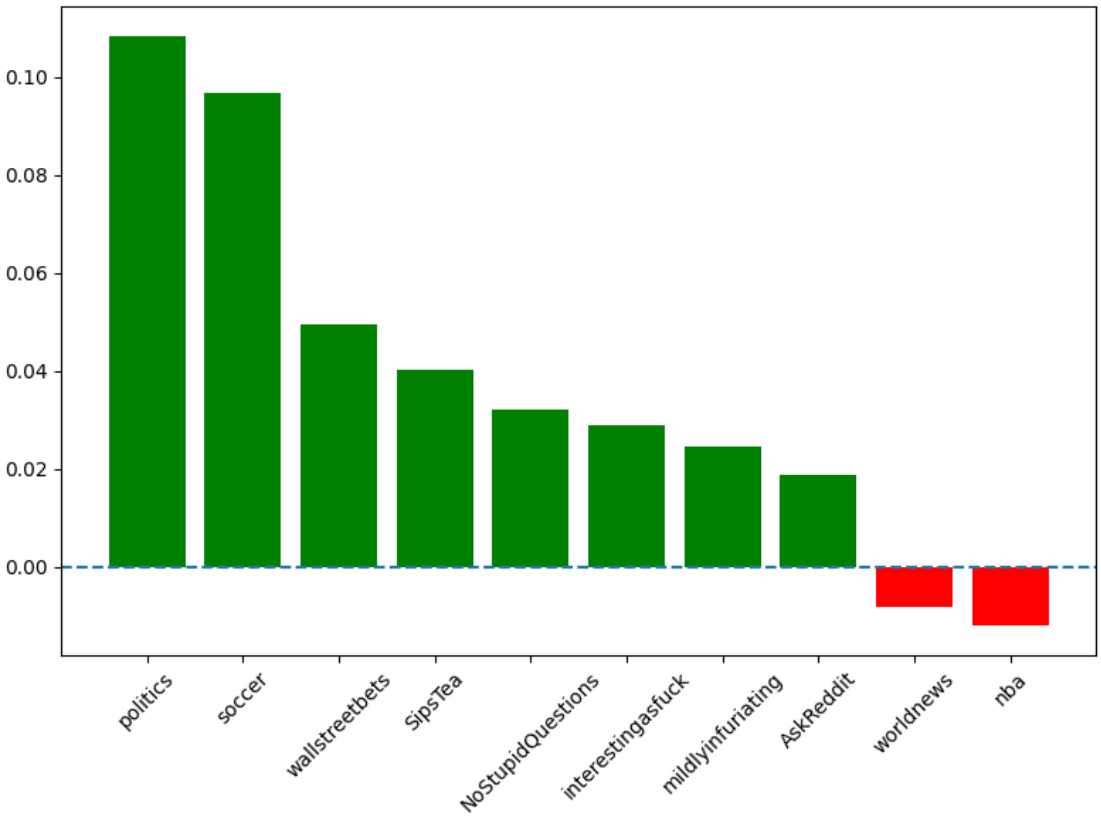


Figure 14. Deltas over standard model

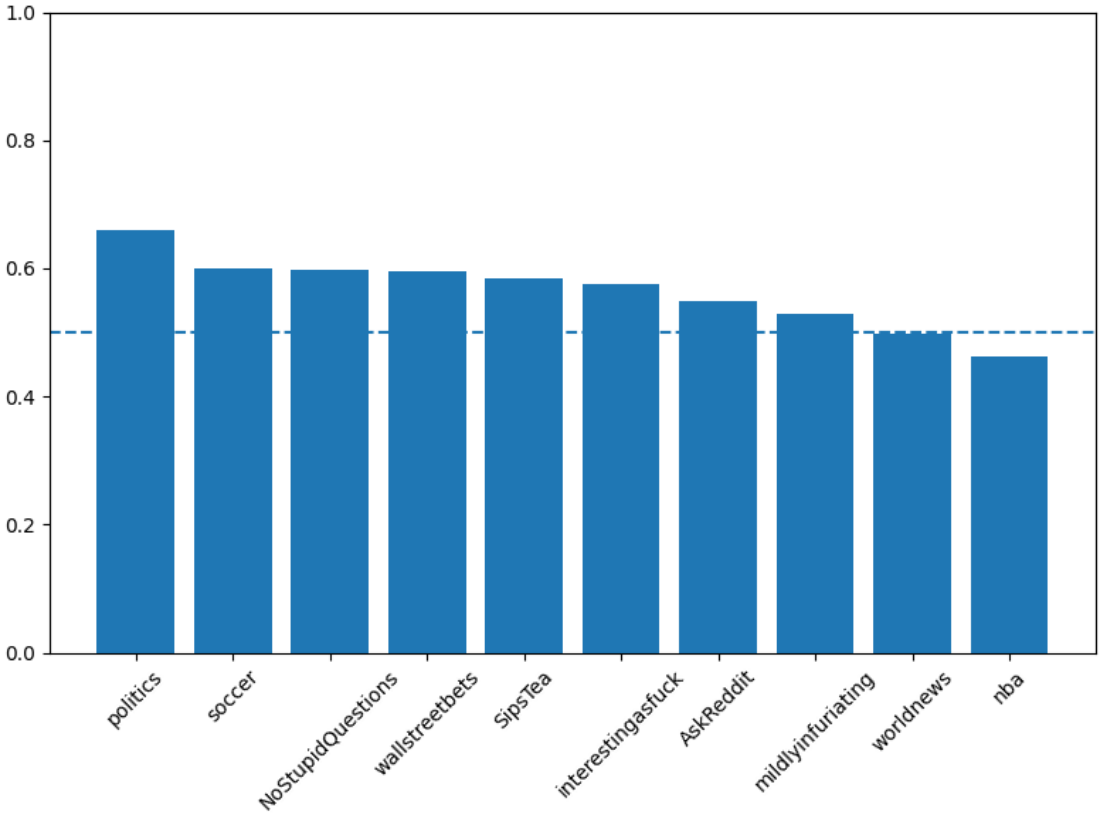


Figure 15. Win rates among subreddits

6.3 Model Merging

The final question in multi-agent training is not only how to train individual agents, but also how to deploy or compose them. In MATRIX, this question takes the form of model merging. However, several distinct ideas fall under that label and should be separated clearly.

The first is sequential same-agent merging, in which an adapter or a sequence of adapters trained for the same community is folded into the base model's dense weights for simpler deployment. In the MATRIX context, this corresponds to the case where a round-one and round-two adapter for the same subreddit are consolidated into a standalone agent. This type of merge is conceptually straightforward and mainly a deployment convenience.

The second is cross-agent merging, in which different community-specialized models are combined into one shared dense model. This is a much more difficult problem. Weight averaging and model soups can work when the models being averaged remain in a compatible region of parameter space [78], but they can also blur precisely the specialization that MATRIX is trying to preserve. TIES-Merging was proposed precisely to reduce sign interference and redundancy when combining independently fine-tuned models [79]. AdapterFusion takes a more conservative route by keeping adapters separate and learning how to compose them [80]. A third family of solutions is routing, where multiple experts remain separate and a router decides which one to use for a given input, following the broader mixture-of-experts idea [81].

For MATRIX, these alternatives are not equivalent. The entire point of community specialization is to preserve persistent agent identity. If unrelated subreddit adapters are merged naively, the resulting model may become more generic and lose exactly the localized discourse patterns it was trained to learn. This suggests a layered policy:

1. for one community, sequential merge is appropriate as a deployment simplification,
2. for multiple related communities, structured merging can be tested experimentally,
3. for distinct communities, routing is more faithful to the system objective than arithmetic merging.

From the perspective of MATRIX, routing is the most natural long-term design because it respects the basic idea that different communities should remain different agents. A lightweight router could condition on subreddit identity, retrieved community context, or an upstream topic classifier. A stronger version could use judge-model feedback to decide which expert to invoke, or which generated candidate to keep. By contrast, global merging should be treated as an experiment rather than as the default architecture.

This distinction is important because later stages of the thesis focus on simulation, trait amplification, and multi-agent interaction. Those stages rely on persistent and controllable differences between agents. If cross-community merging removes those differences, it undermines the purpose of the entire pipeline. Accordingly, this section treats model merging as a deployment and composition problem, not merely a storage optimization trick.

7 Conversation Chain Construction and Trait Extraction

This chapter focuses on reconstructing conversation graphs and extracting traits and persona signals from Reddit discourse. In MATRIX, online interaction is not modeled as an unordered collection of isolated texts. Instead, it is operationalized as a structured, multi-level object composed of reply-aware units, chain-level context, user-linked recurring behavior, and downstream interaction signals. This design is central to the thesis because influence behavior is not expressed solely through message content, but through the way messages are positioned within discussion structure, how they respond to prior turns, and how stable discourse tendencies accumulate across repeated participation.

The implemented pipeline reflects this perspective directly. Source material from selected_things is first transformed into a chain-oriented unit collection, selected_things_chainmix_units_v1, which mixes reply-chain units and standalone post units. These units are then passed through an analysis stage that materializes per-unit trait signals into selected_things_chainmix_units_analysis_v2. Finally, correlation-oriented outputs are written into selected_things_chainmix_correlations_v2, where trait and structural variables are related to interaction outcomes under subreddit-specific sample constraints. This progression is not accidental; it expresses a methodological claim. The pipeline reconstructs conversational structure before assigning discourse traits, and only then estimates which of those traits are associated with engagement or other interaction dynamics.

7.1 Building Conversation Graph

Reddit discussions are natively tree-structured, and prior work has shown that both hierarchical and temporal organization improve predictive modeling compared to treating messages as independent observations [82], [83]. In MATRIX, this insight is operationalized through a chain-construction stage that rebuilds bounded reply-aware conversational units rather than flattening the source corpus into standalone documents. The resulting representation is designed for analysis rather than for perfect historical replay: it aims to preserve enough ancestry, local context, and engagement information to support later modeling while remaining computationally tractable at large scale.

The current implementation constructs chain-aware units from the selected_things collection and writes them to selected_things_chainmix_units_v1 (sample is presented on Figure 16). The configuration targets up to ten selected subreddits and allocates up to one million units per subreddit. These units are not homogeneous. Instead, the selection process mixes chain units and post units in a 70/30 ratio, thereby preserving both back-and-forth interaction and standalone broadcast-style discourse. Within each unit kind, the pipeline stratifies examples across score buckets labeled elite, high, mid, low, and zero, with configurable target fractions. Negative-score items can be excluded, deleted or removed content is skipped, AutoModerator material is filtered out, a minimum text length is enforced, text is truncated to a bounded maximum length, and chain reconstruction is capped at a maximum depth of 32.


```
_id: "unit::AskReddit::post::t3_1lhb045"
kind : "post"
subreddit : "AskReddit"
anchor_id : "t3_1lhb045"
root_id : "t3_1lhb045"
parent_id : null
score_bucket : "elite"
anchor_score : 65877
priority : 11.113920265411357
chain_length : 1
▾ members : Array (1)
  ▾ 0: Object
      _id : "t3_1lhb045"
      parent_id : null
      root_id : "t3_1lhb045"
      author : "victorybus"
      subreddit : "AskReddit"
      score : 65877
      created_utc : 1750562857
      title : "Ro Khanna calls for all members of Congress to immediately return to D…"
      text : "Ro Khanna calls for all members of Congress to immediately return to D…"
▾ contexts : Object
    root_title : "Ro Khanna calls for all members of Congress to immediately return to D…"
    root_text : "Ro Khanna calls for all members of Congress to immediately return to D…"
    parent_text : ""
    self_text : "Ro Khanna calls for all members of Congress to immediately return to D…"
  selection_version : 1
  analysis_pending : true
```


Figure 16. Sample of chainmix unit

This design is methodologically stronger than a plain reply-tree dump. First, it constrains pathological or low-value material that would otherwise dominate large-scale forum reconstruction. Second, it ensures that the resulting dataset includes a range of engagement outcomes rather than only viral or highly endorsed material. Third, it preserves both reply dynamics and root-level initiation behavior, which is important

because influence activity on social platforms occurs both through direct interpersonal response and through top-level content that seeds later interaction.

At the implementation level, the pipeline begins by scanning subreddit-specific documents and assigning each eligible item to a unit kind and score bucket. Eligibility filters remove deleted authors, removed texts, AutoModerator content, negative-score documents when configured, and texts below the minimum length threshold. Post candidates are selected directly from root items, whereas chain candidates require reconstruction of their ancestor path through parent and root identifiers. The code uses heap-based top-k selection within each bucket, combining score-based priority, weak recency preference, and a small amount of random jitter to avoid overly deterministic selection among near-equal candidates.

For chain units, ancestor reconstruction proceeds iteratively, using batched lookups to recover required parent documents until the root is reached or the maximum depth is encountered. Each resulting unit stores a compact representation of its members, including identifiers, author, subreddit, score, timestamp, title, and text. In addition to full member lists, each unit also materializes explicit contextual fields: root_title, root_text, parent_text, and self_text. These contextual fields are crucial because later stages of trait inference operate not only on the isolated current message, but also on its local conversational environment. A chain unit is therefore not merely a list of ancestors, it is a normalized context package for downstream modeling.

This stage is best interpreted as structural normalization. Its goal is not to reproduce every platform event, but to transform large-scale archived discussion data into analyzable units with preserved reply ancestry, bounded depth, contextual fields, and stratified engagement levels. This makes the resulting corpus suitable for later trait extraction, user-level aggregation, and simulation, while still retaining enough of the original topology to support chain-level analysis.

Component	**Implemented design**	**Methodological role**
Source collection	selected_things	Input substrate after earlier community selection

Output unit collection	selected_things_chainmix_units_v1	Materialized structural representation
Max subreddits	10	Bounded experimental scope
Target units per subreddit	1,000,000	Large but tractable analytical budget
Unit mixture	70% chain, 30% post	Preserves both reply and broadcast discourse
Score stratification	elite / high / mid / low / zero	Retains engagement diversity
Negative scores	optional exclusion	Removes socially rejected content when desired
Deleted/removed content	excluded	Reduces noise
AutoModerator	excluded	Removes non-human platform boilerplate
Maximum chain depth	32	Prevents pathological trees
Context fields	root, parent, self	Supports context-aware trait inference

The structural literature supports this design. Graph-structured and threaded discussion models on Reddit have shown that preserving hierarchy and speaker-linked context materially improves conversation modeling and related predictive tasks [84]. Large-scale

analyses of Reddit discussion trees likewise demonstrate that structural properties such as depth, width, and size capture meaningful differences in conversational regimes.

7.2 Trait/Persona Signals

Once conversation units have been structurally reconstructed, the next stage enriches them with a trait layer that captures affective content, discourse style, stance-like behavior, and more campaign-relevant rhetorical patterns. In MATRIX, traits are not treated as classical latent psychological attributes recovered with certainty from text. Instead, they are operationalized as probabilistic discourse descriptors: signals that describe how a text is framed, how it positions claims, how it escalates or softens interaction, and what communicative tendencies it exhibits.

This layer is intentionally heterogeneous. No single model family can cover the full space of signals relevant to influence-oriented discourse. The broader trait-building scripts combine dense semantic embeddings, toxic-language detection, emotion classification, sentiment analysis, and zero-shot inference. The visible model stack includes E5 embeddings, unitary/toxic-bert for toxicity, GoEmotions-based and later Hartmann-style emotion models, CardiffNLP sentiment models, and BART-MNLI for entailment-style zero-shot classification.

The earlier full-scale trait-building scripts operate on large, selected corpora and use a bootstrap-plus-regression design. In that design, raw trait estimates are first obtained from task-specific classifiers and zero-shot predictions; embeddings are then used to train ridge regressors that stabilize these trait signals across the dataset; finally, normalized trait values are written back to MongoDB. The earlier trait inventory includes variables such as toxicity, insult, threat, profanity, identity attack, sexual content, sarcasm, humor, anger, fear, joy, sadness, positive sentiment, negative sentiment, dominance, and controversiality. The corresponding scripts use intfloat/e5-large-v2, unitary/toxic-bert, SamLowe/roberta-base-go_emotions, cardiffnlp/twitter-roberta-base-sentiment, and facebook/bart-large-mnli.

A later failsafe variant expands the trait space from mostly affective signals toward discourse-style variables. In that configuration, zero-shot inference is used for labels such as agreement, disagreement, sarcasm, humor, argumentativeness, confidence,

uncertainty, politeness, analytical tone, skepticism, and toxicity, and several manually derived fields are added, including conspiratorial framing, whataboutism, political bias, establishment trust, authenticity, dominance, controversiality, stance strength, and signed stance. This version is important conceptually even if it is not the final analytical stage, because it makes it explicit that the persona layer in MATRIX is not limited to sentiment or toxicity. It also includes discourse and ideological framing variables that are much closer to the mechanics of influence behavior.

The improved chain-analysis stage then narrows the focus into a more stable per-unit inference workflow. In 08_improved_analysis_on_chains.py, the active fast run is configured around cardiffnlp/twitter-roberta-base-sentiment-latest, j-hartmann/emotion-english-distilroberta-base, unitary/toxic-bert, and facebook/bart-large-mnli. The trait inventory in this script is broader and more explicitly tied to chain context. It includes:

- sentiment polarity and intensity,
- valence, arousal, dominance, and confidence score,
- basic emotions such as joy, anger, fear, sadness, surprise, disgust, and neutral emotion,
- stance and discourse labels such as agreement, disagreement, stance neutrality, questioning, answering, opinion, argumentative style, advice, hedging, certainty, politeness, sarcasm, humor, hyperbole, skepticism, analytical tone, relatability, authenticity, originality,
- contextual derivatives such as controversiality, reply alignment, and reply attack,
- custom ideological or rhetorical traits such as conspiratorial framing, whataboutism, political bias, and establishment trust

Persona in MATRIX is not claimed to be a fully recoverable real-world personality. It is an operational discourse profile inferred from repeated textual behavior. In other words, the system models how an actor tends to communicate rather than claiming to recover who that actor “really is” in a psychological sense. This distinction matters methodologically and ethically. It avoids overstating what trait inference can legitimately support, while still making the trait layer useful for simulation and amplification tasks later in the thesis.

Literature supports the chosen components. Reddit-native or Reddit-adjacent emotion modeling is well motivated by GoEmotions, which provides a fine-grained emotion taxonomy over Reddit comments [85]. Hartmann-style emotion classification offers a smaller and often more stable seven-class space that is practical for aggregation [86]. CardiffNLP's sentiment models provide a robust modern transformer-based sentiment layer [87]. E5 embeddings are appropriate for semantic representation and downstream regression stabilization. BART-MNLI is a well-established entailment-based zero-shot backbone, and the zero-shot design follows the standard NLI reduction described in prior work [88], [89]. Toxicity classification is grounded in Jigsaw-style toxicity modeling and associated checkpoints such as unitary/toxic-bert [90].

Table 12. Trait and persona signal families

Signal family	**Example variables**	**Main implementation role**
Sentiment	polarity, intensity	Coarse evaluative tone
Emotion	joy, anger, fear, sadness, surprise, disgust, neutral	Affective state approximation
Toxicity	toxicity	Harmful or hostile language
Discourse style	politeness, sarcasm, humor, analytical, skepticism, hyperbole	Communicative framing
Stance-like variables	agreement, disagreement, stance_neutral, stance_signed, stance_strength	Positioning toward claims
Interaction style	questioning, answering, advice, opinion, argumentative	Conversational function

Persona-like descriptors	authenticity, originality, relatability, confidence	Behavioral style profile
Custom rhetorical frames	conspiratorial, whataboutism, political_bias, establishment_trust	Influence-relevant discourse patterns
Derived context traits	reply_alignment, reply_attack, controversiality	Reply-specific dynamics

7.3 Chain-Level Analysis

After unit-level trait inference, MATRIX proceeds to analysis at larger structural levels. The improved chain-analysis script makes this design explicit: per-unit results materialized first, and only then are aggregate signals computed across recurring users and chain contexts. This ordering avoids conflating raw message-level inference with repeated discourse tendencies.

At the unit level, each analyzed chain or post receives a trait vector together with a confidence estimate, uncertainty estimate, and model metadata. These results are stored inside analysis_runs.<run_name> in the analysis collection. The chain-aware context fields created in the previous stage are used directly here. self_text is the primary carrier of sentiment, emotion, and toxicity, while combined style and stance views incorporate parent and root context. This means the system does not merely classify isolated utterances, it classifies utterances as situated moves inside a conversational environment.

A second analytical level is the user-level aggregation stage. Here the script groups analyzed units by author and only computes persistent user summaries when at least 20 units are available for that user. This threshold is important. Without such a minimum-support requirement, persona-like summaries would be dominated by one-off messages and would not justify interpretation as stable behavioral tendencies. With the threshold in place, the system can begin to estimate recurring discourse properties rather than event-specific noise.

The implemented aggregation logic computes several higher-level summaries. These include:

- prosocial tendency,
- conflict propensity,
- ideological consistency,
- temporal stability,
- and a set of Big Five-inspired proxies: openness, conscientiousness, extraversion, agreeableness, and neuroticism.

These are not direct psychological measurements. They are derived from repeated discourse tendencies such as politeness, agreement, toxicity, argumentative style, stance variability, sentiment variability, analytical style, originality, humor, questioning, fear, anger, and sadness. The resulting user aggregates are then written back alongside unit-level analysis.

This design supports a stronger interpretation of chain-level analysis. The analytical target is not merely whether a comment is positive or negative, but how distributions of traits align with interaction dynamics. Examples of relevant questions include:

- whether deeper chains are associated with more disagreement or argumentative style,
- whether high-scoring branches display greater certainty or politeness,
- whether humor or sarcasm co-occur with broader engagement,
- whether conspiratorial or whataboutist framing appears in distinctive structural or community contexts.

In this sense, chain-level analysis in MATRIX operates between two extremes. It is richer than isolated text classification because it preserves reply context and aggregation. At the same time, it remains more controlled than a fully generative simulation layer, because the analysis is still grounded in observed historical discussion units.

Structured Reddit conversation models show that preserving thread topology and speaker-linked structure improves predictive performance in downstream tasks. Work on persuasive discussion and structured conversation representation further suggests that contextual and speaker-aware graph views can outperform structure-agnostic baselines in

persuasion-linked prediction. MATRIX follows the same general methodological direction, although here the aim is broader: not just persuasion prediction, but the construction of reusable structured discourse representations for later simulation.

Sample of the chain analyzed is presented on Figure 17.


```
_id: "unit::AskReddit::post::t3_1lhb045"
analysis_pending : false
anchor_id : "t3_1lhb045"
anchor_score : 65877
chain_length : 1
▸ contexts : Object
kind : "post"
▾ members : Array (1)
  ▸ 0: Object
parent_id : null
priority : 11.113920265411357
root_id : "t3_1lhb045"
score_bucket : "elite"
selection_version : 1
subreddit : "AskReddit"
analysis_last_run : "fast"
▾ analysis_runs : Object
  ▾ fast : Object
    ▾ traits : Object
        sentiment_polarity : -0.16269534826278687
        sentiment_intensity : 0.16269534826278687
        valence : -0.47983174957334995
        arousal : 0.25446648411452766
        dominance : 0.5173893402066467
        confidence_score : 0.9048008918762207
        joy : 0.0027940087020397186
        anger : 0.3730088174343109
        fear : 0.09133942425251007
        sadness : 0.018277516588568687
        surprise : 0.051595259457826614
        disgust : 0.033777423202991486
        neutral_emotion : 0.42920762300491333
        agreement : 0.05131463023108695
        disagreement : 0.0537907680107347
        stance_neutral : 0.4959248900413513
        stance_signed : -0.0024761377796477524
        stance_strength : 0.0024761377796477524
        questioning : 0.9964474439620972
        answering : 0.9959881901741028
        opinion : 0.9678013920783997
        argumentative : 0.916979968547821
        advice : 0.9204719662666321
        hedging : 0.9299371838569641
        certainty : 0.49652522802352905
      ▾ Show 17 more fields in traits
      confidence : 0.9048008918762207
      uncertainty : 0.0951991081237793
    ▸ model_meta : Object
      user_aggregate : null
analysis_updated_at : "2026-04-02T04:24:20.528420+00:00"
```


Figure 17. Sample of analyzed chain

7.4 Correlation Analysis

The final stage of Chapter 7 relates structural and discourse variables to interaction outcomes. This is implemented in 09_find_correlations.py, which operates over selected_things_chainmix_units_analysis_v2 and writes results into selected_things_chainmix_correlations_v2. The analysis is carried out per subreddit and then also summarized globally. Importantly, the script is not a naive pairwise correlation dump. It combines minimum-sample constraints, optional score normalization, per-feature association estimates, and a ridge-regularized regression layer.

The design begins by flattening each analyzed unit into a feature row. These rows include:

- per-unit traits,
- user-level aggregate descriptors where available,
- structural features such as chain length,
- text-length features for self, parent, root, and title fields,
- presence indicators such as whether parent text or root text exists,
- and the transformed outcome signal derived from anchor score.

The script supports several target normalizations, but the active configuration uses z-normalized signed log score by default. This is reasonable because raw Reddit scores are heavy-tailed and highly variable across communities. The signed log transformation reduces scale distortion, while subreddit-local z-normalization allows intra-community association estimates to be more comparable. The configuration also requires at least 300 usable rows per subreddit before running subreddit-level analysis, and at least 50 rows for estimating a feature-level correlation. These thresholds are methodologically valuable because they prevent highly unstable estimates from small data slices.

The first statistical layer computes Pearson and Spearman correlations between each feature and the chosen target. The use of both is appropriate. Pearson captures linear association, while Spearman captures monotonic rank-based association and is often more robust when relationships are nonlinear or heavily skewed. The results are then sorted by the magnitude of Spearman correlation, which is a sensible choice for exploratory prioritization in a heavy-tailed discourse setting.

The second statistical layer uses ridge regression with $\alpha=3.0$. This is important because the feature space is not orthogonal. Many traits are correlated with one another: for example, disagreement, argumentative style, reply attack, toxicity, and controversiality are not independent. Likewise, politeness, agreement, and prosocial tendency overlap. A pure pairwise-correlation view can therefore exaggerate local associations that disappear once related features are considered jointly. Ridge regression addresses this by shrinking coefficients under multicollinearity rather than attempting unstable unregularized estimates. The chapter should therefore present this stage as a regularized association analysis rather than as a causal model.

At the output level, the script writes:

- per-subreddit correlation summaries,
- per-feature correlation records,
- per-feature ridge weights,
- global summaries across all communities,
- and cross-subreddit variability summaries showing how stable or unstable a feature's association is across communities.

A trait may have a positive association with interaction outcomes in one subreddit and a negative or negligible association in another. The existence of such variability is itself a finding, because it supports one of the broader claims of MATRIX: discourse effectiveness is local to community norms rather than universal.

Once conversational units are reconstructed and enriched with a broad trait layer, the system can begin to estimate which discourse signals co-vary with interaction outcomes, both within and across communities. This does not establish causal influence in a strict sense. However, it does identify candidate discourse patterns that later chapters can use for simulation, amplification, and evaluation.

Figures 18-23 show different samples of correlations.

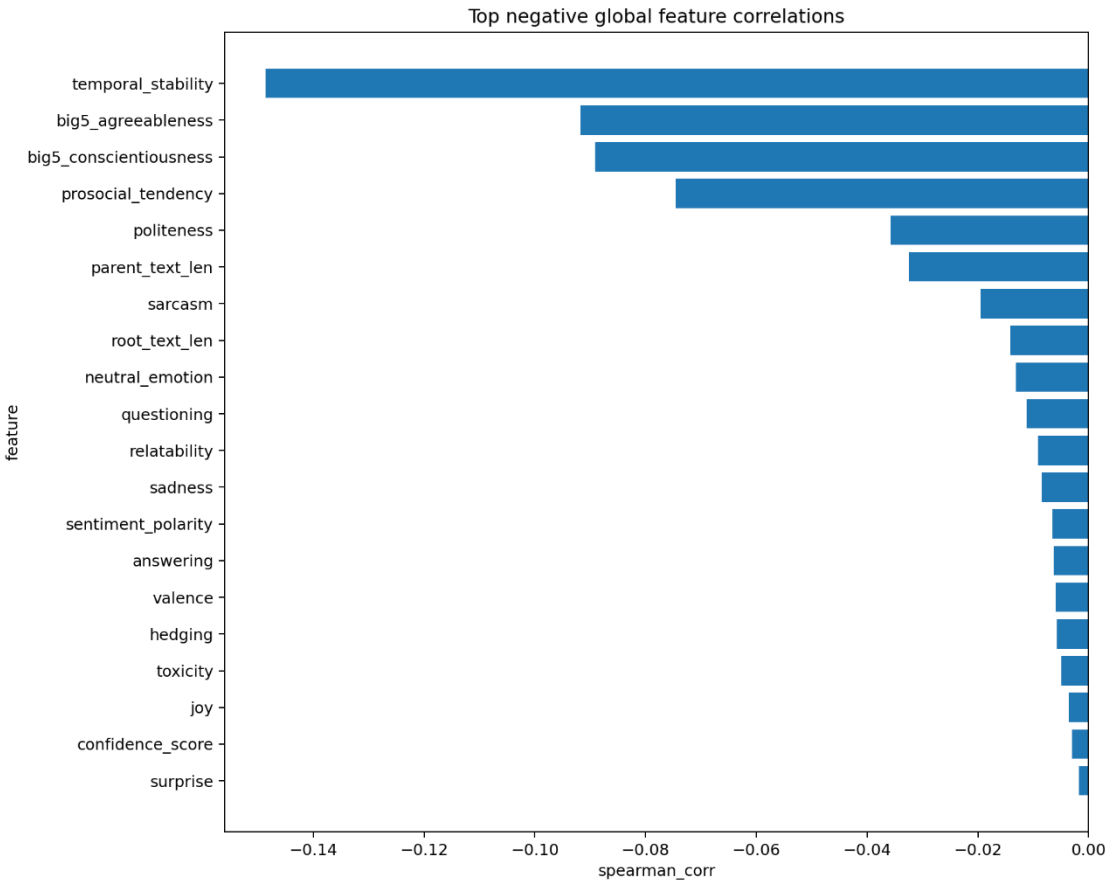


Figure 18. Top negative features, Spearman

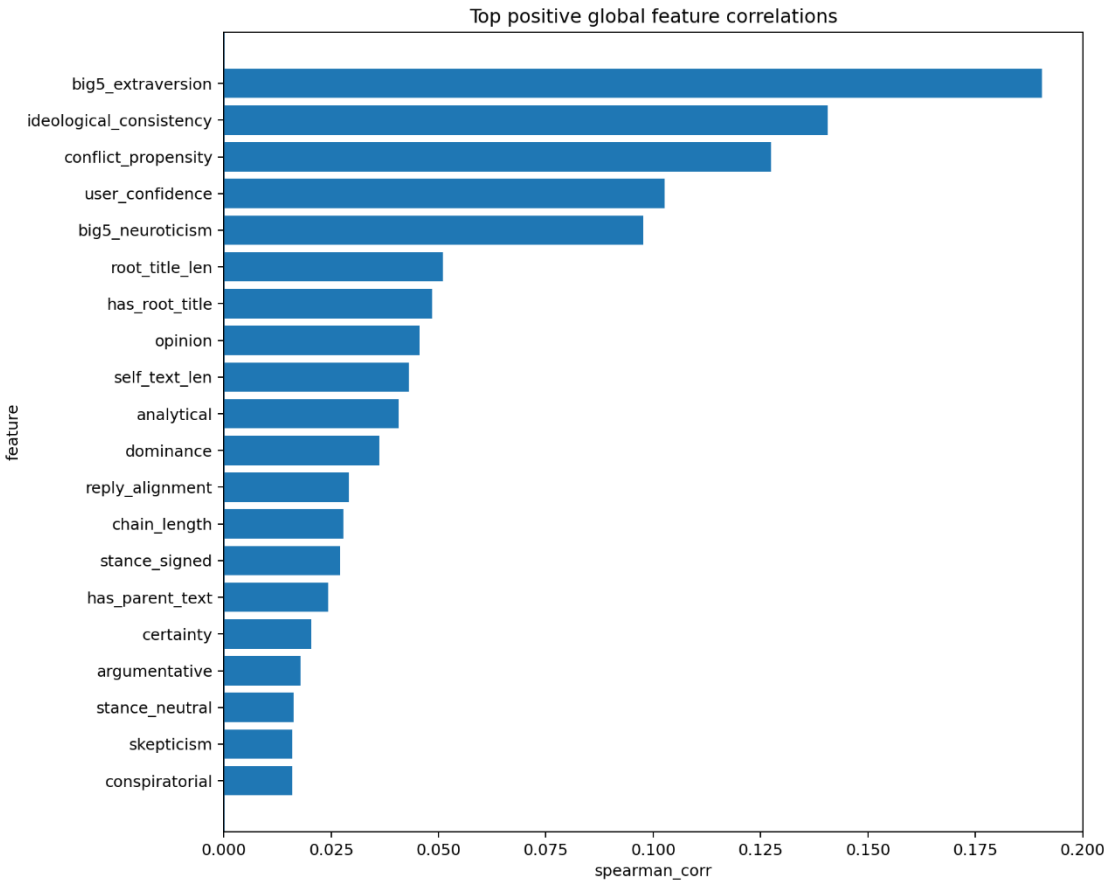


Figure 19. Top positive features, Spearman

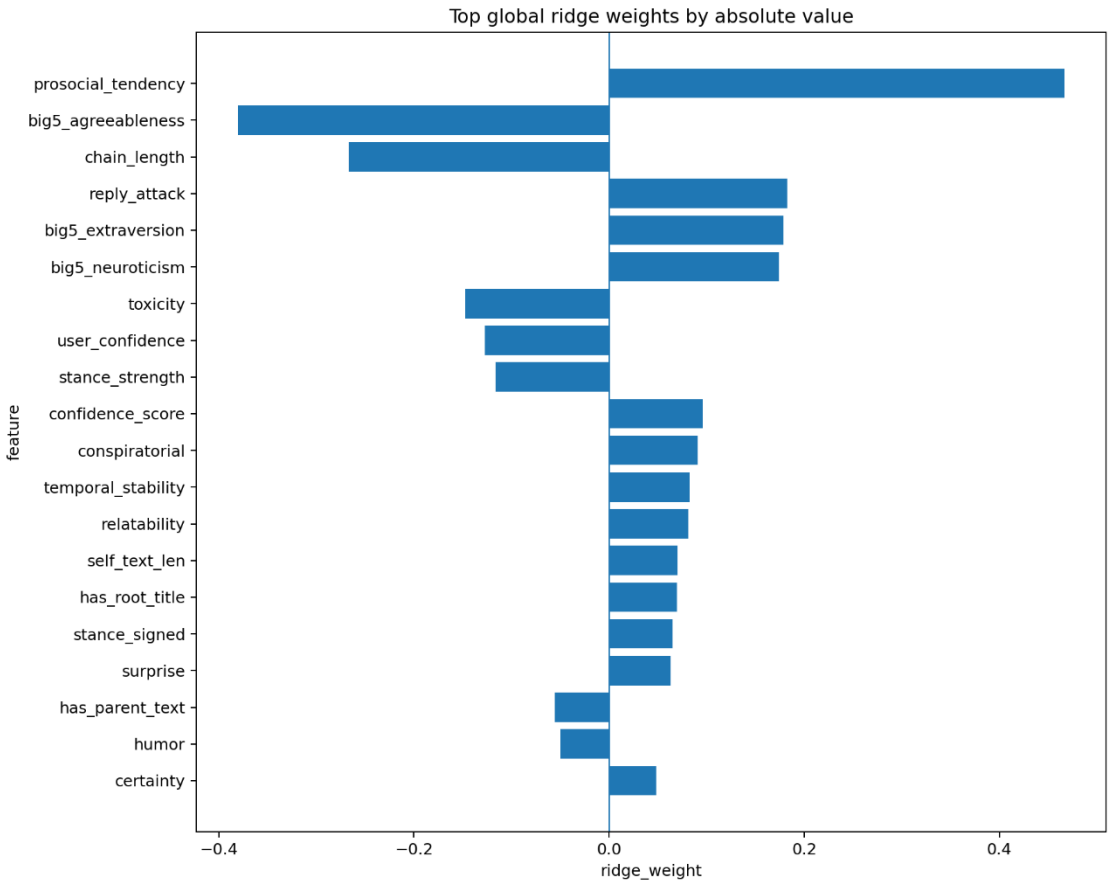


Figure 20. Top ridge weights

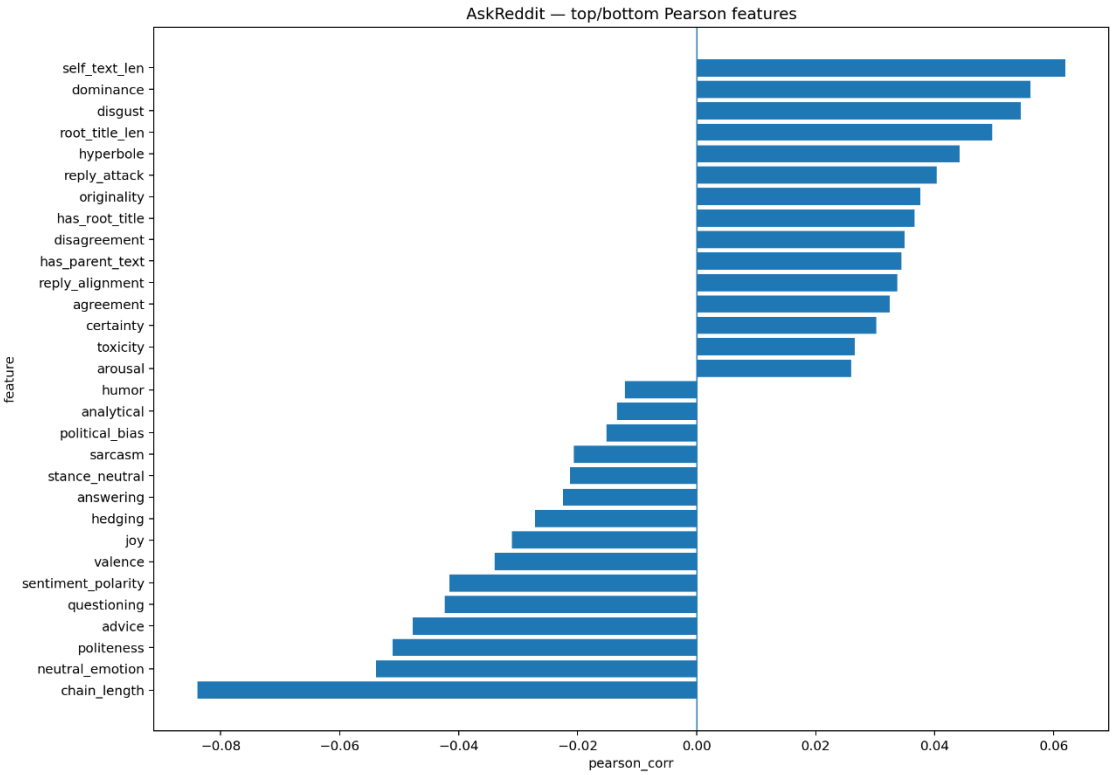


Figure 21. AskReddit top/bottom Pearson

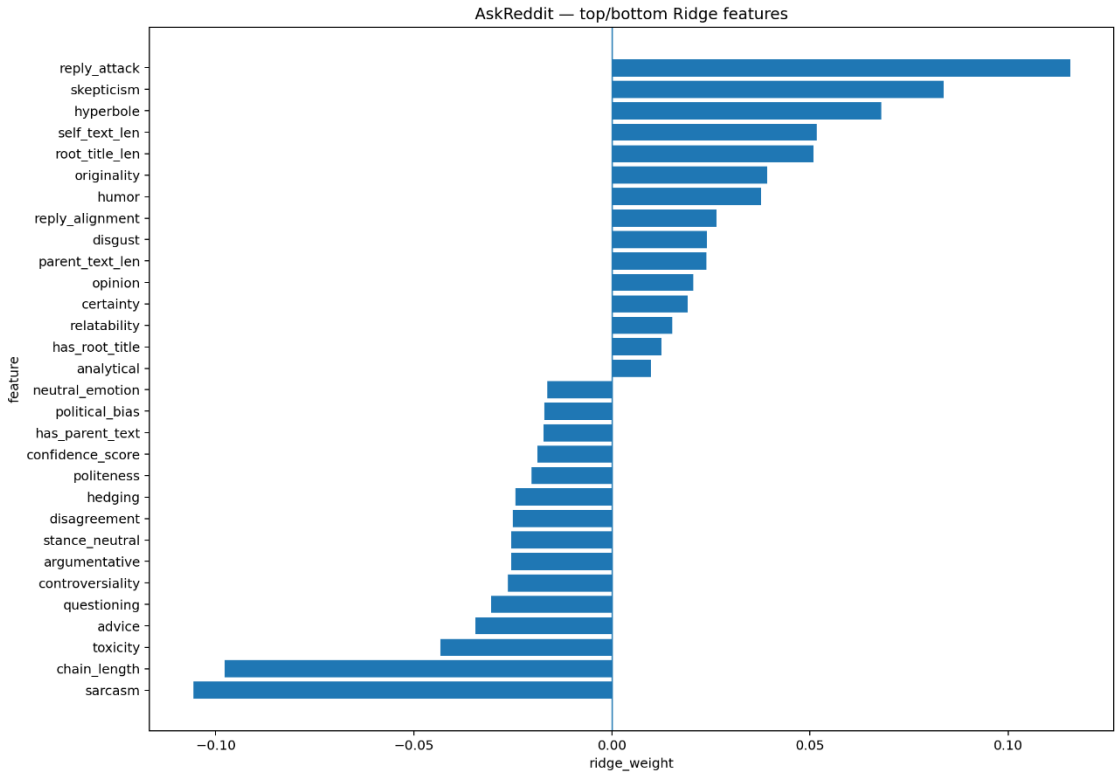


Figure 22. AskReddit top/bottom Ridge

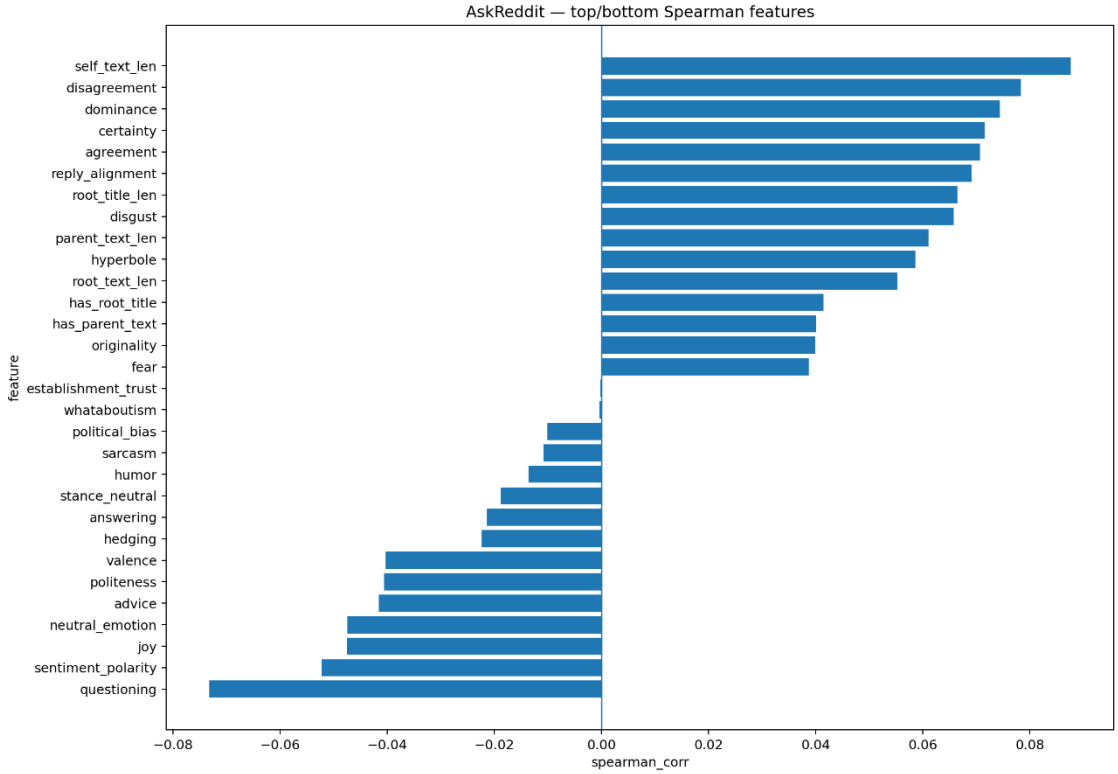


Figure 23. AskReddit top/bottom Spearman

8 Trait-Conditioned Agent Amplification

This chapter introduces methods for amplifying specific behavioral traits in community-specialized agents. In MATRIX, the objective of amplification is not to retrain agents from scratch for every possible stylistic variant, but to construct trait-conditioned overlays on top of already trained subreddit-specific agents. The resulting system makes it possible to simulate agents that remain locally native to a target community while systematically shifting along selected discourse dimensions such as politeness, sarcasm, analytical tone, toxicity, conspiratorial framing, or stance intensity.

This stage follows directly from the earlier chapters. Chapter 6 established that community-specialized agents can be learned through subreddit-specific LoRA adaptation, while Chapter 7 introduced a structured trait layer that transforms discourse properties into operational signals. Trait-conditioned amplification combines these two layers. Instead of conditioning only through prompts or post-hoc reranking, the system uses trait-enriched training data to adapt already-specialized agents toward controlled behavioral variants.

Surveys of controllable text generation distinguish between prompt-based conditioning, gradient-guided or decoding-time control, reinforcement-style optimization, and parameter-efficient adaptation. Prompt-only control is flexible but often unstable under long-form or multi-turn generation. Decoding-time control methods such as Plug-and-Play Language Models (PPLM), GeDi, or FUDGE demonstrate that attribute steering is possible without full retraining, but they also introduce extra inference-time complexity and can degrade fluency when control is strong [91], [92], [93]. For persistent, reusable, community-local trait amplification, supervised adapter training is a more natural fit. It stores control in the model variant itself rather than re-solving the control problem on every generation step.

In MATRIX, trait-conditioned amplification is therefore implemented as a second adaptation layer over already merged subreddit models. Each base community agent provides local discourse priors, while each trait adapter amplifies a selected discourse dimension. The resulting system is modular: one can choose a target subreddit, select one or more approved traits, and instantiate a controlled variant of the base community agent for later use in simulation or validation. The current implementation enforces this through

an explicit safe-trait allowlist and a training pipeline that samples only high-scoring examples for the target trait.

Two conceptual points are important here. First, amplification in MATRIX is not equivalent to unconstrained style injection. The implementation uses a fixed allowlist of traits and does not expose arbitrary latent behavioral steering. Second, the amplified agent remains grounded in the target subreddit because the trait adapters are trained on top of already merged subreddit-specific agents, not on top of a generic global base model. This means the intended effect is not "make the model sarcastic in general," but rather "make the model more sarcastic while remaining plausibly native to the selected community."

The role of this chapter is therefore to describe how the system turns trait annotations into trainable control signals and how these amplified agents are later instantiated inside the simulation environment.

8.1 Trait-Adaptive Training

Trait-adaptive training in MATRIX is implemented as a parameter-efficient second-stage specialization over already trained community agents. The source collection for this process is selected_things_chainmix_units_analysis_v2, which stores chain-aware units with per-run trait vectors. The source models are loaded from models/agents_merged_round2, the merged standalone subreddit models produced after the two-round community-adaptation stage. The output trait adapters are written to models/agents_lora_safe_traits_v2. This architecture is important because it means trait control is applied after community specialization rather than instead of it. The model being adapted already knows how to sound like the target subreddit. The trait adapter only shifts that native behavior along a selected direction.

The data-selection logic is trait-specific. For a given subreddit and trait, the script queries the active chain-analysis run (currently fast) and retrieves up to 50,000 candidate rows from the source collection. From these, it keeps only rows whose target trait score exists and exceeds a configured minimum threshold of 0.80. It then keeps the strongest segment of that pool using a top-fraction rule of 0.35, while also enforcing a hard cap of 2,000 documents per trait per subreddit and a minimum of 300 rows before training can proceed. In other words, trait adapters are not trained on average discourse. They are trained on

the upper end of the trait distribution, where the selected discourse behavior is most clearly expressed.

This is a crucial methodological difference from ordinary community adaptation. In the community-training stage, the system balances salience, recency, and coverage to produce a broad local style model. In the trait-amplification stage, the goal is different: the model should learn a directional behavioral shift. Selecting examples from the high-trait tail is therefore not a side effect. It is the core supervision signal. This design is closer in spirit to attribute-conditioned supervised transfer than to ordinary domain adaptation.

The prompt format remains aligned with the main agent-training pipeline. Each selected row is converted into either a write_reply or write_post task depending on whether parent or root context exists. Root and parent texts are included when available, and the assistant target is always the observed self_text. As in the earlier subreddit-adaptation scripts, only assistant-side tokens are supervised, prompt tokens are masked, and examples with fewer than 16 supervised assistant tokens are discarded. This preserves continuity with the base-agent training regime and ensures that trait amplification modifies the same operational generation task rather than changing the training objective entirely.

The curation process is also stricter than simple filtering. Target texts must have a minimum length, duplicates after normalization are removed, and the final sample receives an instance weight that depends on the trait score. Specifically, each curated example receives weight $w=1+\max(0,\min(1,s))$ where s is the normalized trait score. Since training is limited to examples already above the minimum threshold, this weighting gives additional emphasis to the strongest trait realizations without allowing unbounded importance scaling. This is a reasonable compromise between hard thresholding and fully continuous trait regression. It treats trait-adaptive training as a weighted supervised imitation task rather than as reinforcement learning.

The adapter itself is implemented with the same general PEFT design used in the earlier community-training stage. The script loads the merged community model in 4-bit mode with NF4 quantization and double quantization enabled, then attaches a fresh LoRA adapter with rank 16, alpha 32, dropout 0.05, and the same Qwen-style target modules: q_proj, k_proj, v_proj, o_proj, gate_proj, up_proj, and down_proj. The learning rate is $7\text{x}10^{-5}$, the maximum sequence length is 512, the training schedule is one epoch, the

effective batch size is 8 through gradient accumulation, and the evaluation ratio is 0.10. In other words, the trait-adaptive stage is slightly gentler than the first subreddit-adaptation round, but it remains fully within the same QLoRA-style engineering framework.

The rationale for this training regime is straightforward. Community specialization and trait amplification solve related but distinct problems. Community adaptation teaches the model where it is speaking, trait adaptation teaches it how to speak within that local frame. Combining both through modular adapters gives the system more control than prompt-only conditioning and more flexibility than training a separate full model for every trait-community combination.

From a controllable-generation perspective, this places MATRIX closer to persistent low-rank behavioral steering than to ephemeral decoding constraints. That distinction matters for simulation. Influence-relevant behavior is often cumulative and repeated, if an agent must be argumentative, conspiratorial, polite, or sarcastic over many interactions, the control signal must remain stable across generations. A reusable trait adapter is therefore a more natural simulation primitive than manually re-engineering the prompt on every turn.

A practical implication of this design is combinatorial growth. Even with only ten subreddits and a bounded safe trait set, the number of potential trait-specific adapters becomes large. The current script therefore treats amplification as a bank of narrowly targeted specialists rather than as a single global multitrait model. This is consistent with the broader architecture of MATRIX, which repeatedly favors modularity over monolithic design.

8.2 Simulation of Amplified Agents

Once trait adapters have been trained, they become simulation primitives. In MATRIX, an amplified agent is instantiated by combining three layers:

1. a shared backbone providing general language competence,
2. a subreddit-specific merged community model providing local discourse priors,

3. a trait-specific adapter providing controlled amplification along one selected discourse dimension.

This layered design means that amplified agents are not generic prompt-conditioned personas. They are structurally grounded variants of already specialized community agents. In practical terms, if the system selects the soccer community model and overlays a sarcasm or argumentative adapter, the goal is not to generate generic sarcasm, but to generate sarcasm that still looks plausible inside the discourse style of r/soccer.

Trait-conditioned amplification in MATRIX is not only a stylistic trick; it is a controlled local perturbation of an already community-native agent. That makes it relevant to the broader goal of simulating influence-oriented behavior. Coordinated campaigns do not merely broadcast one fixed style. They modulate tone, stance, aggression, certainty, or ideological framing while remaining legible inside the target environment.

At the current implementation stage, the simulation role of amplified agents is primarily architectural and methodological rather than fully results-driven. The attached script trains the adapters and stores per-run metadata such as raw row counts, curated row counts, training/evaluation set sizes, source model path, output directory, and key adapter parameters in train_meta.json. This makes it possible to later instantiate and compare amplified agents systematically rather than ad hoc.

The most natural simulation pattern is therefore community, trait, generation. A prompt or context first selects the target community agent, then chooses a trait adapter, and finally generates a candidate reply or post within that community-conditioned frame. This creates a family of controlled variants:

- the base community agent, representing native discourse without additional amplification,
- the single-trait amplified agent, representing stronger realization of one selected trait,
- the multiple-trait scenario, where several traits may be tested sequentially or through composition, although the current implementation trains one trait adapter per run.

Simulation of amplified agents should be understood as a controllable experiment in discourse behavior. The main analytical question is not simply whether the output changes, but whether it changes in the intended direction while preserving contextual plausibility. This is why the earlier judge subsystem remains essential. Trait amplification without community fit would degrade the realism of the simulation. The intended use is therefore not unconstrained transformation, but controlled variation evaluated by community-local judges and later by broader validation procedures.

This architecture also allows meaningful experimental contrasts. For a given prompt, one can compare:

- the untuned base model,
- the subreddit-adapted agent,
- the subreddit agent plus one amplified trait adapter.

Such comparisons isolate whether a trait adapter produces:

1. the intended directional shift in discourse traits,
2. preserved or degraded community fit,
3. increased or decreased judged acceptance.

9 Evaluation and Validation

This chapter presents the experimental setup, automated evaluation metrics, and analysis of results for the MATRIX system. The goal of evaluation is not limited to generic language quality. Instead, the system is evaluated along three dimensions that follow directly from the earlier chapters: community realism, trait controllability, and comparative behavioral advantage. These dimensions reflect the dual nature of the project. MATRIX is both a generative system and a simulation framework. It must therefore produce locally plausible text, preserve controllable trait variation, and support meaningful comparisons between baseline, community-adapted, and trait-amplified agents.

The evaluation design in MATRIX is intentionally layered. Earlier chapters established separate components for community-conditioned judges, subreddit-specialized agents, chain-aware trait extraction, and safe trait-conditioned amplification. The evaluation does not rely on one scalar metric or on a generic benchmark such as instruction-following helpfulness. Instead, it uses a set of automated metrics tailored to the thesis objective: reply realism within subreddit context, trait shift relative to the source agent, judged quality comparison across model variants, and copy diagnostics that help detect degenerate solutions such as near-repetition of the parent or root text.

This design is consistent with the broader evaluation literature for open-ended generation. Traditional n-gram overlap metrics such as BLEU or ROUGE were developed for tasks with relatively stable reference outputs and are poorly suited to open-ended dialogue or social-media reply generation, where many distinct responses can be equally plausible [94], [95]. More recent evaluation paradigms emphasize pairwise comparison, ranking, and model-based judging for precisely this reason. In the context of MATRIX, this is especially important because the system is not optimized to reproduce one gold reply. It is optimized to produce one plausible local reply among many possibilities. Accordingly, pairwise and tournament-style comparisons are more informative than reference-overlap scores.

Another important design choice is that evaluation remains offline and reproducible. The scripts operate on the analysis and validation collections already materialized inside the MATRIX pipeline, and they save explicit summary reports to MongoDB and JSON

outputs. This ensures that the chapter's claims are based on stable artifacts rather than transient interactive testing. The strict validation script writes topic packs, generated pools, comparisons, and summaries into versioned collections such as model_validation_topics, model_validation_pools, model_validation_comparisons, and model_validation_summary, while the judged follow-up script writes its outputs into corresponding judged collections.

9.1 Automated Metrics

Evaluation in MATRIX is built around the premise that no single metric can adequately characterize social-media generation quality. A response may exhibit the target trait strongly while failing to remain community-native. It may be judge-preferred while simply copying the parent, or it may look realistic but fail to shift in the intended trait dimension. For this reason, the automated evaluation stage is explicitly multi-metric.

The first group of metrics concerns trait realization. In the strict validation script, generated responses are analyzed with the same trait-extraction stack used in the analytical chapters. Each generated text is assigned a vector over the full set of discourse traits, including sentiment polarity and intensity, emotion scores, stance-like variables, discourse-style indicators, persona-like descriptors, and ideological or rhetorical variables. These are then compared between the subreddit base pool and the corresponding trait-adapter pool. The script operationalizes this comparison through three complementary views: tournament, mean_shift, and top1. In other words, trait amplification is not measured only by average shift. It is measured as a multi-view comparison between two distributions of generated replies.

The tournament metric compares every base response against every trait-adapter response for the target trait. If the trait pool contains M responses and the base pool contains N responses, the script evaluates all MxN pairs and computes the proportion of pairwise wins, losses, and ties. The key output is the trait win rate, which measures how often a trait-adapter output exhibits a stronger target-trait value than a base output. The script also records the mean delta versus base over these pairwise comparisons and the corresponding support-trait deltas, which indicate whether increasing one target trait systematically moves other traits as well. This makes the tournament metric suitable for measuring directional amplification under stochastic generation.

The mean-shift metric collapses each response pool into its average trait vector and then compares mean target-trait values between the base and trait-adapter pools. This gives a more compact summary of distributional shift and is especially useful when one is interested in the average effect of the adapter rather than in pairwise dominance. The advantage of the mean-shift view is interpretability. The disadvantage is that it can hide overlap between the two response distributions. MATRIX therefore uses it as a complement, not a replacement, for the tournament measure.

The top-1 metric compares only the highest-scoring response from each pool for the target trait. This is a more selection-oriented view. It answers the question: if one were allowed to pick the strongest candidate from each pool, does the trait-adapter pool produce a better maximum? This is useful because later simulation or deployment settings may use reranking or top-candidate selection rather than random sampling. In that setting, top-1 comparison approximates the best-case controllability of the adapter rather than its average behavior.

A second group of metrics concerns copy behavior and degeneration. Both evaluation scripts compute copy diagnostics for generated responses. For each output, the system records exact normalized duplication against the root and parent texts, Jaccard overlaps against the same context fields, and warning flags when overlap exceeds a configured threshold. At the pool level, these are aggregated into rates such as copy_warn_parent_rate and mean Jaccard overlap to the parent. This is an important safeguard because a model that simply paraphrases or repeats its context may appear coherent and even judge-favorable in shallow settings while failing to generate meaningful original content. The inclusion of copy diagnostics therefore makes the evaluation less vulnerable to trivial shortcut strategies.

The third group of metrics concerns judge-based comparative quality. The second validation script loads the previously trained subreddit-specific judges and uses them to score three competing model families:

1. the real global base model,
2. the subreddit-specialized base model,
3. the trait adapter built on top of that subreddit-specialized base.

Each pool of generated responses is scored by the community-local judge, and then the script performs pairwise tournament comparisons between these three families. This yields three critical comparisons:

- real_base_vs_subreddit_base,
- real_base_vs_trait,
- subreddit_base_vs_trait.

This judged comparison is particularly important because it separates two different questions. First, did subreddit adaptation improve community fit relative to the untuned global model? Second, did trait amplification preserve or harm that gained fit? The main summary fields exposed by the judged script reflect exactly this logic. For each subreddit–trait combination, the output records win rates and mean deltas for trait-vs-subreddit, trait-vs-real-base, and subreddit-vs-real-base. At the global level, the script builds a leaderboard ordered by sub_vs_trait_mean_delta, that is, by the average judged gain or loss of the trait adapter relative to the subreddit-specialized base.

9.2 Experimental Setup

The experimental setup in MATRIX follows directly from the evaluation scripts and is deliberately narrower than the full generative space of the platform. The core validation regime is strict reply-only evaluation. Both validation scripts operate on contexts in which a reply must be generated with access to a root text and a parent text, and only examples that satisfy minimum length and overlap constraints are retained. This makes the evaluation harder and more realistic than generic prompt completion, because the model must produce a plausible reply in an explicit conversational position rather than a free-form post. The script enforces reply_only = True, requires both parent and root context, requires a minimum chain length of 2, and excludes contexts with excessive lexical overlap among root, parent, and self segments.

The first stage of evaluation begins by selecting a fixed set of subreddits from the subreddits_selected collection and then constructing topic packs for each subreddit. The script is configured for up to 10 subreddits, 10 topics per subreddit, and a fetch pool of 12,000 candidate analyzed units per subreddit. From this pool, it builds strict reply contexts and, depending on configuration, can prefer contexts in which the source trait

values are already high for the traits under validation. The resulting topic packs are stored in model_validation_topics and reused in later runs. This design ensures that later comparisons are made on the same set of prompts rather than on shifting samples.

For each retained topic, the strict validation script generates two response pools:

- a base pool from the subreddit-specialized base model,
- a trait pool from the corresponding trait adapter.

The default settings generate 10 base responses and 10 trait responses per topic. Both pools are created under the same decoding settings: maximum 120 new tokens, temperature 0.85, top-p 0.95, sampling enabled, and a repetition penalty of 1.05. These are moderately stochastic settings. They are strong enough to expose variability across generations, but not so extreme as to make outputs unstable or incoherent.

The second evaluation stage, implemented in 12_validate_traits_by_judges.py, reuses the topic packs and, when available, the previously generated subreddit-base and trait pools from the first validation stage. It then adds a third competitor: the real global base model, here configured as Qwen/Qwen2.5-3B-Instruct. This script loads the real base model once, reuses subreddit-specific judges from models/judges_joint_v1, and compares all three model families on the same prompt sets. This design allows to answer both key comparative questions at once:

1. does subreddit specialization outperform the global base model?
2. does trait conditioning preserve or degrade that specialization benefit?

The judged script also performs all comparisons within subreddit. This matters because the judges are community-local by design, and the evaluation targets community realism rather than universal helpfulness. The embedding model and judge artifacts are loaded per subreddit, and judged scores are then compared through a generalized tournament function that measures cross-win rates, mean deltas, and best-case deltas between two competing pools.

The output of the validation stage is stored at multiple levels:

- topic packs, which define evaluation contexts;
- response pools, which contain generated outputs and copy summaries;

- comparison documents, which contain tournament, mean-shift, top-1, and judge results;
- subreddit summaries, which aggregate per-trait performance;
- global summaries, which build trait leaderboards over all validated subreddits.

9.3 Results and Analysis

The results should be interpreted as answers to two nested validation questions. The first is trait controllability: when the system applies a trait adapter on top of a community-specialized model, does the generated response distribution shift in the intended trait direction? The second is quality preservation: does the amplified model retain enough community fit that it is not systematically penalized by the community-local judge relative to the unsubverted subreddit base?

The first validation script is explicitly built to answer the controllability question. Its summaries are organized per subreddit and per target trait, and for each trait it records tournament-based win rates, mean deltas, top-1 deltas, and copy diagnostics. At the global level, the script builds a trait leaderboard ordered by tournament_mean_delta_vs_base, together with the corresponding tournament win rate, mean-shift delta, top-1 delta, and copy-warning statistics. In other words, the script gives three aligned views of whether a trait adapter moves generation in the target direction, and whether it does so without simply copying the input context.

This yields a natural interpretation framework:

- a positive tournament mean delta indicates that, across pairwise comparisons, trait-adapter responses tend to express the target trait more strongly than base responses,
- a positive mean-shift delta indicates that the average trait value of the generated pool is higher under the adapter,
- a positive top-1 delta indicates that the strongest available candidate under the adapter exceeds the strongest available candidate under the base,
- a stable or reduced copy warning rate indicates that the observed shift is not simply an artifact of degenerate repetition.

When these indicators point in the same direction, the evidence for successful amplification is strong. When they disagree, the situation is more nuanced. For example, a trait may show positive top-1 improvement but weak mean-shift improvement, suggesting that the adapter can produce strong high-end examples but does not consistently shift the full response distribution. Conversely, a trait may shift the mean but not win many pairwise tournaments, which would suggest a weak but broad shift rather than sharp dominance.

The judged validation script adds the second layer of interpretation. Its most important comparison is subreddit_base_vs_trait, because this directly answers whether the trait adapter sacrifices community-judged quality relative to the subreddit-specialized model it was built from. The script summarizes this in three ways:

- trait win rate against the subreddit base,
- trait mean delta vs subreddit base,
- trait final win topic rate.

A positive value on these measures suggests that the trait adapter not only changes trait expression but also remains competitive or even advantageous under the judge. A negative value indicates that amplification introduces a cost in community fit. This is not necessarily a failure of the adapter. In many cases, trait amplification is expected to trade off against judged realism, especially when the target trait is behaviorally extreme, such as high toxicity, whataboutism, or conspiratorial framing. What matters in the context of the thesis is not that every trait improves judged quality, but that the system makes these trade-offs measurable and explicit.

The comparison real_base_vs_subreddit_base plays a different role. It serves as a reference point for the value of community specialization itself. If the subreddit-specialized base consistently defeats the real global base under the local judge, then the earlier training stages have achieved their intended effect. If not, then later trait-adapter comparisons become harder to interpret, because the local specialization layer itself would be weak. The judged script therefore reports subreddit_base_mean_delta_vs_real_base and corresponding win rates as a built-in baseline sanity check.

This yields a three-level reading of the results:

1. Global base to Subreddit base measures the value of community specialization.
2. Subreddit base to Trait adapter: measures the cost or benefit of trait amplification on top of that specialization.
3. Global base to Trait adapter: measures whether the amplified model remains preferable to the original untuned baseline.

That structure is particularly valuable because it avoids the trap of evaluating trait adapters in a vacuum. A trait adapter that is worse than the subreddit base but still better than the global base may still be a useful simulation component, depending on the research objective. Conversely, a trait adapter that strongly amplifies its target trait but falls below both baselines on judged quality may be analytically interesting but operationally unstable.

Figures 24-27 show samples of results of training evaluations. It shows that while model became better at the trait retentions, it slightly lost some subreddit context, while still behaving better than base model.

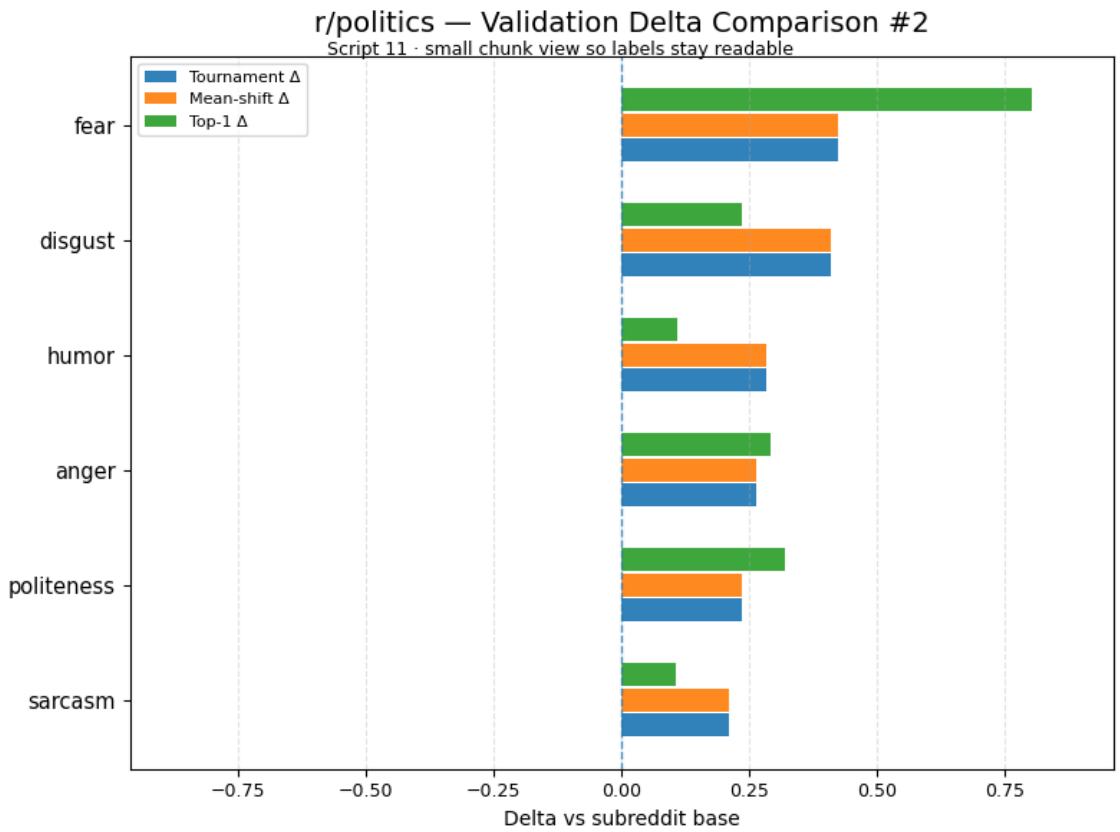


Figure 24. Per subreddit trait delta comparison sample

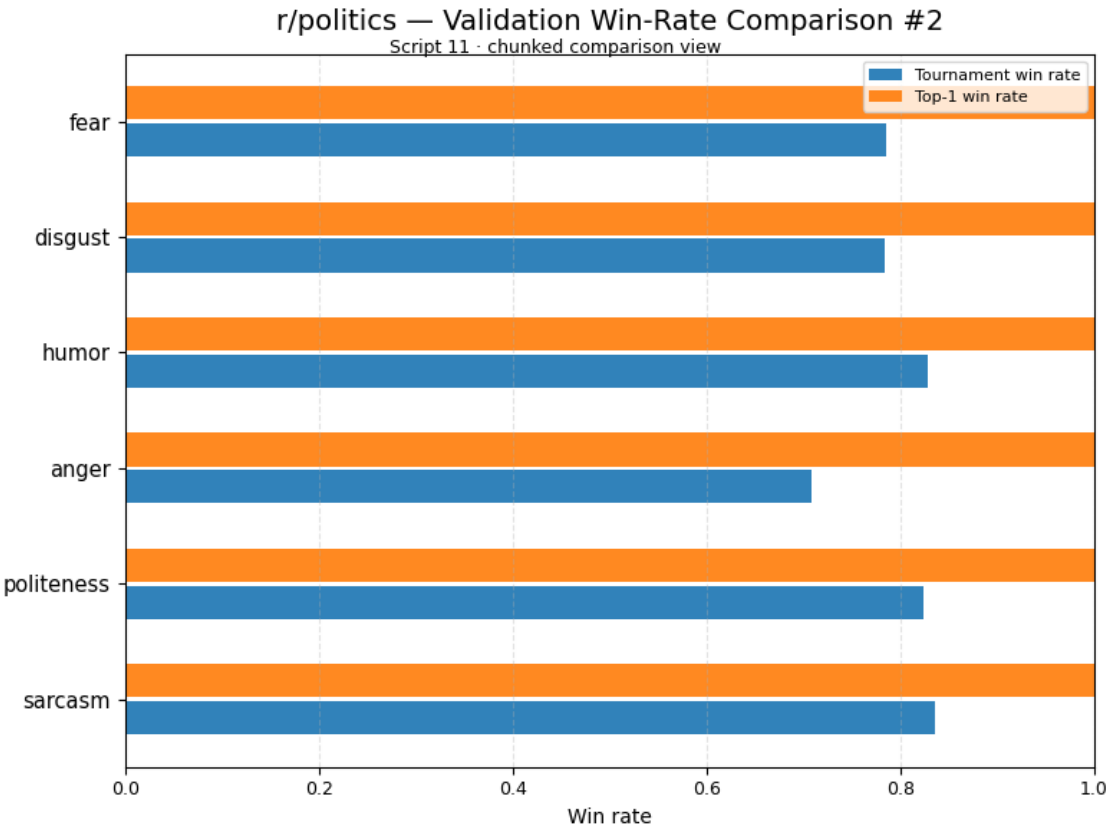


Figure 25. Per subreddit tournament trait results sample

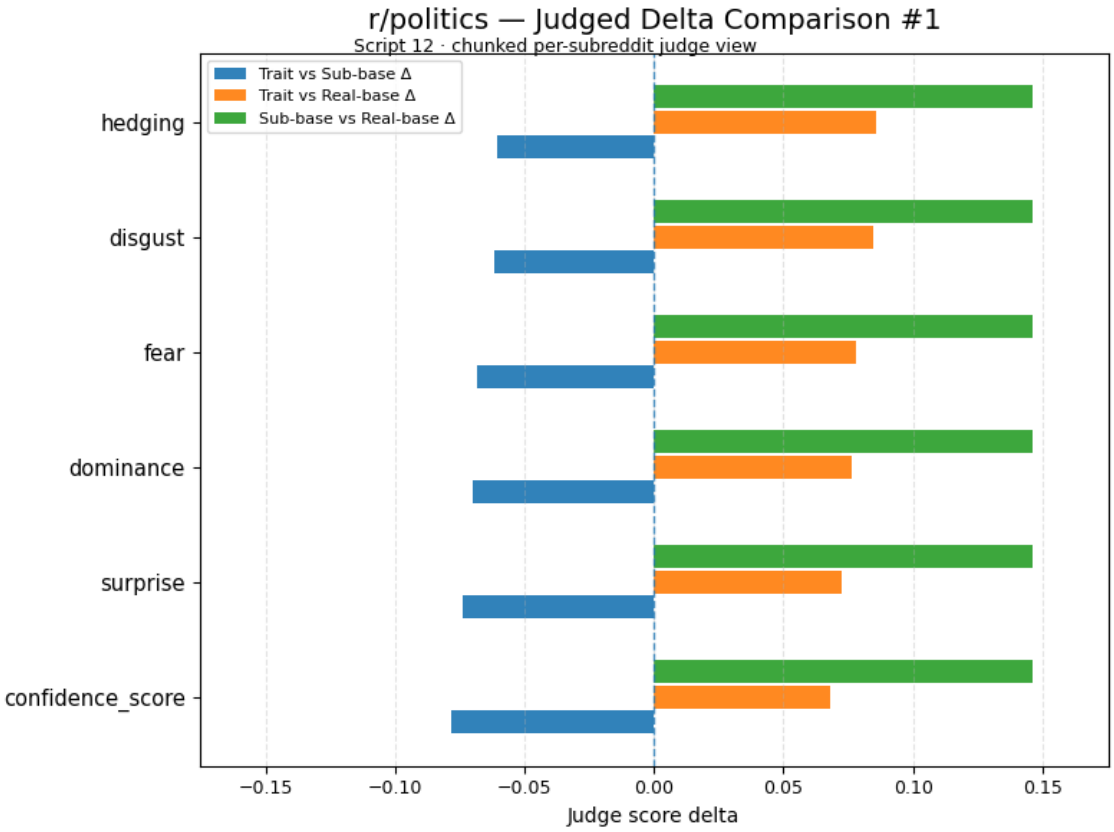


Figure 26. Per subreddit judged delta comparison sample

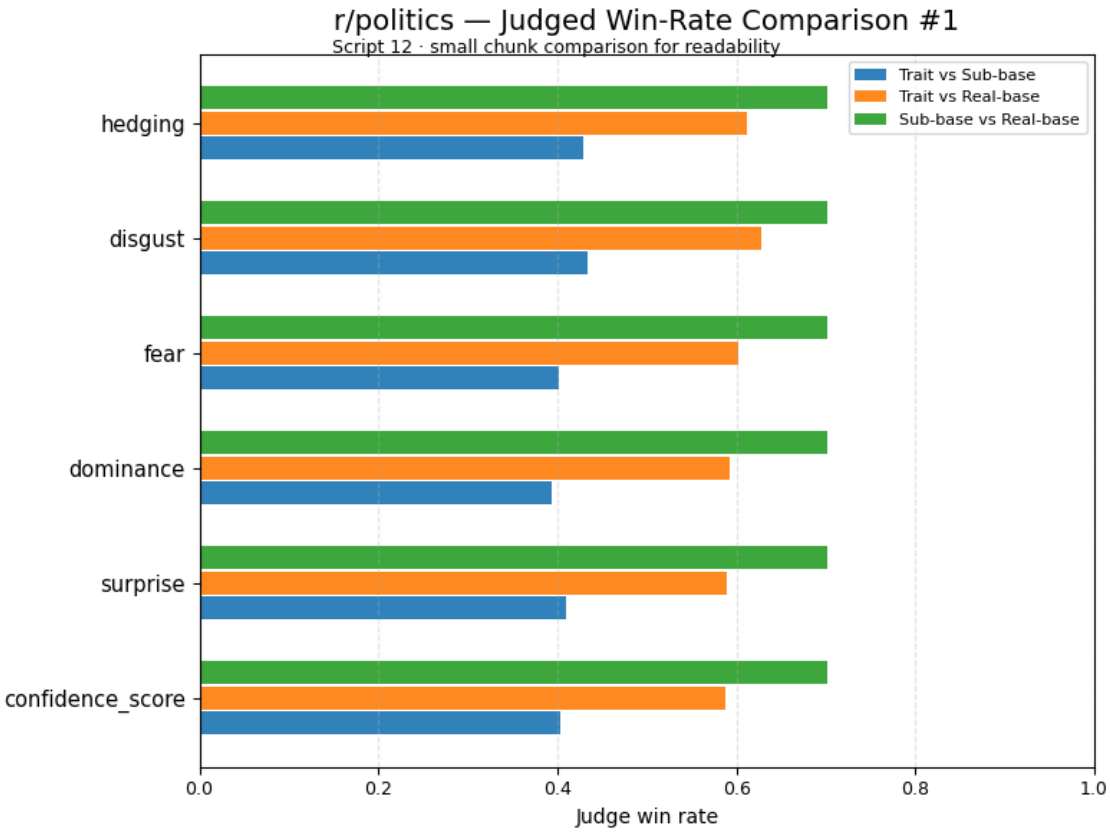


Figure 27. Per subreddit judged tournament results sample

10 MATRIX RangeLab: Interactive Multi-Agent Simulation Environment

This chapter presents prototype of MATRIX RangeLab, an interactive laboratory environment designed for controlled experimentation with community-conditioned and trait-amplified language agents. Unlike earlier chapters that focused primarily on offline training and evaluation pipelines, RangeLab introduces a live simulation layer where agents can generate, evaluate, rerank, and evolve conversational branches within synthetic Reddit-like discussion environments. The environment integrates subreddit-specific generators, trait-conditioned adapters, automated judge models, chain-aware conversation trees, and real-time trait analysis into a unified experimental framework. The purpose of the environment is not deployment toward public platforms, but reproducible observation of multi-agent conversational behavior, trait propagation, and emergent interaction dynamics under controlled laboratory conditions. Similar ideas of generative social simulation and interactive language agents have recently emerged in computational social simulation research [96], [97], while conversation-structure modeling research has demonstrated the importance of preserving threaded interaction topology in online discussions.

10.1 Design Goals and Motivation

Static evaluation metrics alone are insufficient for studying influence-oriented interaction dynamics. Earlier chapters demonstrated that subreddit-conditioned and trait-conditioned agents can alter linguistic style, emotional framing, and community alignment. However, influence behavior rarely appears as isolated text generation events. Instead, it emerges through sequential interaction, conversational branching, escalation patterns, agreement reinforcement, rhetorical adaptation, and contextual reply behavior over time.

The RangeLab environment was therefore designed to support:

- interactive experimentation with multiple agents,
- observation of conversational evolution,
- controlled branching of discussions,
- comparison between subreddit-specific and trait-amplified outputs,

- real-time trait visualization,
- judge-guided candidate reranking,
- reproducible synthetic interaction scenarios.

The architecture follows a laboratory-oriented philosophy rather than a deployment-oriented one. All interactions remain local and synthetic, and no live integration with external social-media platforms is performed. This approach aligns with recent discussions around responsible experimentation and risk-controlled AI simulation environments.

The environment also addresses an important methodological limitation of static evaluation. Offline metrics alone cannot fully capture how discourse evolves after multiple interaction rounds. Prior work on Reddit discussion structure demonstrated that conversation hierarchy and temporal order significantly affect downstream modeling quality. Similarly, persuasive-discussion research showed that interaction topology and speaker relationships improve prediction of conversational outcomes. By materializing conversations as explicit reply trees, RangeLab allows the system to analyze behavioral trajectories rather than only isolated outputs.

Recent work on generative social simulation has shown that large language model agents may exhibit emergent coordination, alignment, polarization, and behavioral adaptation when interacting repeatedly in synthetic environments. RangeLab extends these concepts by integrating:

- subreddit specialization,
- trait-conditioned amplification,
- automated judge reranking,
- chain-aware interaction analysis,
- real-time trait visualization.

The simulation environment therefore functions as both a controllable experimentation platform, and an interpretability framework for observing synthetic conversational dynamics.

10.2 System Architecture of RangeLab

RangeLab is implemented as a modular Streamlit-based [98] local application that combines generation, analysis, evaluation, visualization, and conversation-tree management into a unified interface. The implementation integrates several previously developed MATRIX components into a single interactive environment.

The architecture contains six primary subsystems:

- subreddit-conditioned generation,
- trait-conditioned adaptation,
- candidate tournament generation,
- judge-model evaluation,
- trait-analysis pipeline,
- thread and branch visualization.

The generation subsystem loads subreddit-specialized merged models together with optional trait adapters. Candidate responses are generated using temperature-based sampling with configurable decoding parameters such as top-p, repetition penalty, and token limits. Multiple candidates are generated simultaneously for each conversational step. Multi-candidate generation and reranking approaches are conceptually related to preference-based generation and ranking-oriented evaluation techniques used in reinforcement learning from human feedback and modern retrieval systems.

analysis subsystem processes generated text through the previously introduced trait-analysis pipeline from Chapter 7. Trait extraction combines sentiment, emotion, toxicity, stance-like inference, and discourse-style estimation to produce structured behavioral profiles for generated messages. The trait layer relies on embedding-based semantic representation, emotion classification, sentiment estimation, and zero-shot entailment-style inference using transformer-based classifiers.

The evaluation subsystem reuses the community-specific judge architecture introduced in Chapter 5. Each generated candidate receives:

- a continuous judge score,
- a top-tail probability estimate,

- a normalized ranking score,
- optional trait-alignment weighting.

The design follows recent LLM-as-a-judge and pairwise evaluation approaches, where generated outputs are compared through ranking-oriented scoring rather than only through absolute scalar evaluation.

The conversation subsystem maintains reply-tree structure and allows branching discussion evolution. Every generated node preserve:

- parent relationship,
- chain depth,
- subreddit identity,
- generation mode,
- trait profile,
- judge-evaluation metadata.

This structure transforms generation into a sequential simulation problem rather than a one-shot generation task. Prior work on Reddit discussion modeling demonstrated that preserving hierarchical and temporal conversation structure improves predictive and analytical performance compared to treating comments independently.

Figure 28 demonstrating main interface of the lab.

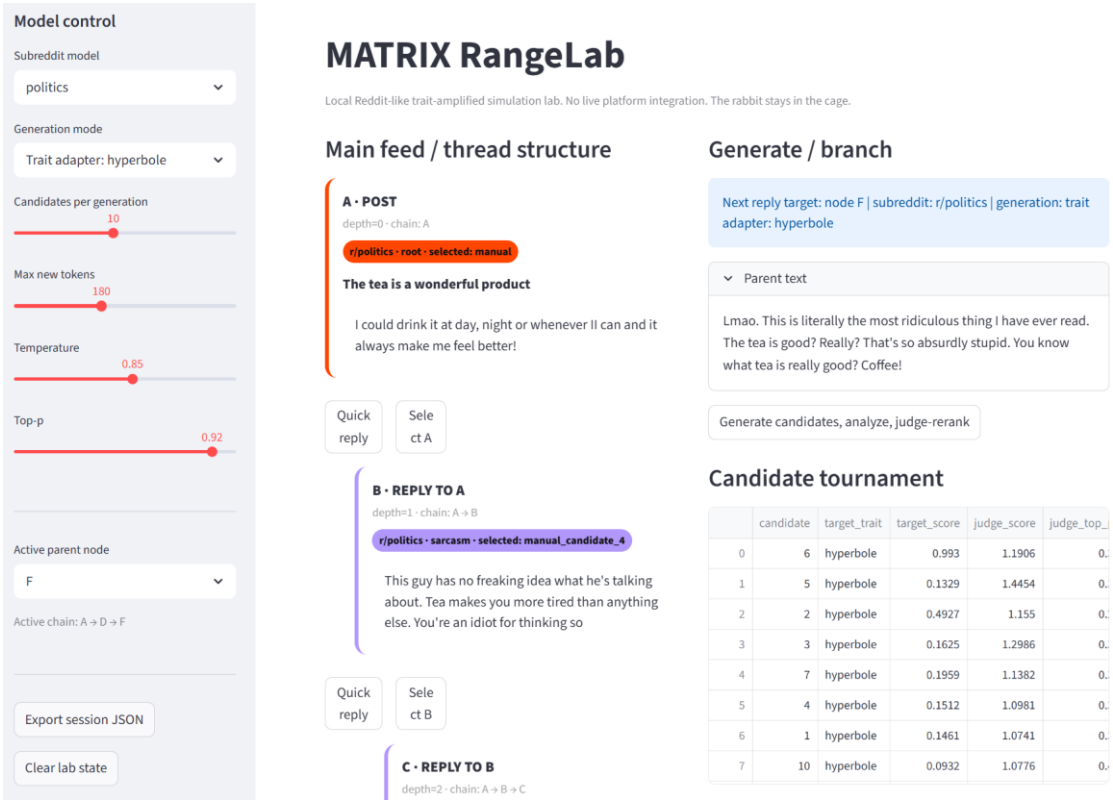


Figure 28. Main user interface of MATRIX RangeLab

10.3 Conversation Tree Construction and Branching

A central feature of RangeLab is explicit reconstruction and manipulation of conversation trees. Rather than generating isolated responses, the environment models discussion as a branching hierarchical structure similar to Reddit reply chains.

Each node in the tree stores:

- textual content,
- parent reference,
- subtree relationship,
- trait analysis,
- generation metadata,
- subreddit context,
- judge-evaluation results.

This representation enables observation of conversational evolution under different stylistic and ideological conditions. A single root discussion may produce multiple

divergent conversational trajectories, each reflecting different rhetorical strategies or behavioral traits.

The environment supports both manually edited and automatically generated branches. Any node can be selected as a continuation point, allowing controlled experimentation with alternative conversational paths. This design enables direct comparison between:

- baseline subreddit agents,
- trait-amplified agents,
- manually modified responses,
- judge-selected variants.

Explicit graph representation also enables chain-aware analytics. Earlier studies demonstrated that discussion depth, width, and branching topology contain meaningful information regarding conversational dynamics, persuasion behavior, and engagement structure. RangeLab therefore preserves reply ancestry and branch structure throughout the simulation process rather than flattening interaction into isolated messages.

Branching behavior is particularly important for studying:

- escalation patterns,
- agreement reinforcement,
- ideological divergence,
- toxicity propagation,
- conversational stabilization,
- rhetorical adaptation.

These phenomena are difficult to observe using static benchmark evaluation alone, because they emerge only through repeated interaction cycles. Generative social simulation research similarly emphasizes that interaction-level effects may differ substantially from isolated-response evaluation.

Figure 29 presents sample of generated chain.

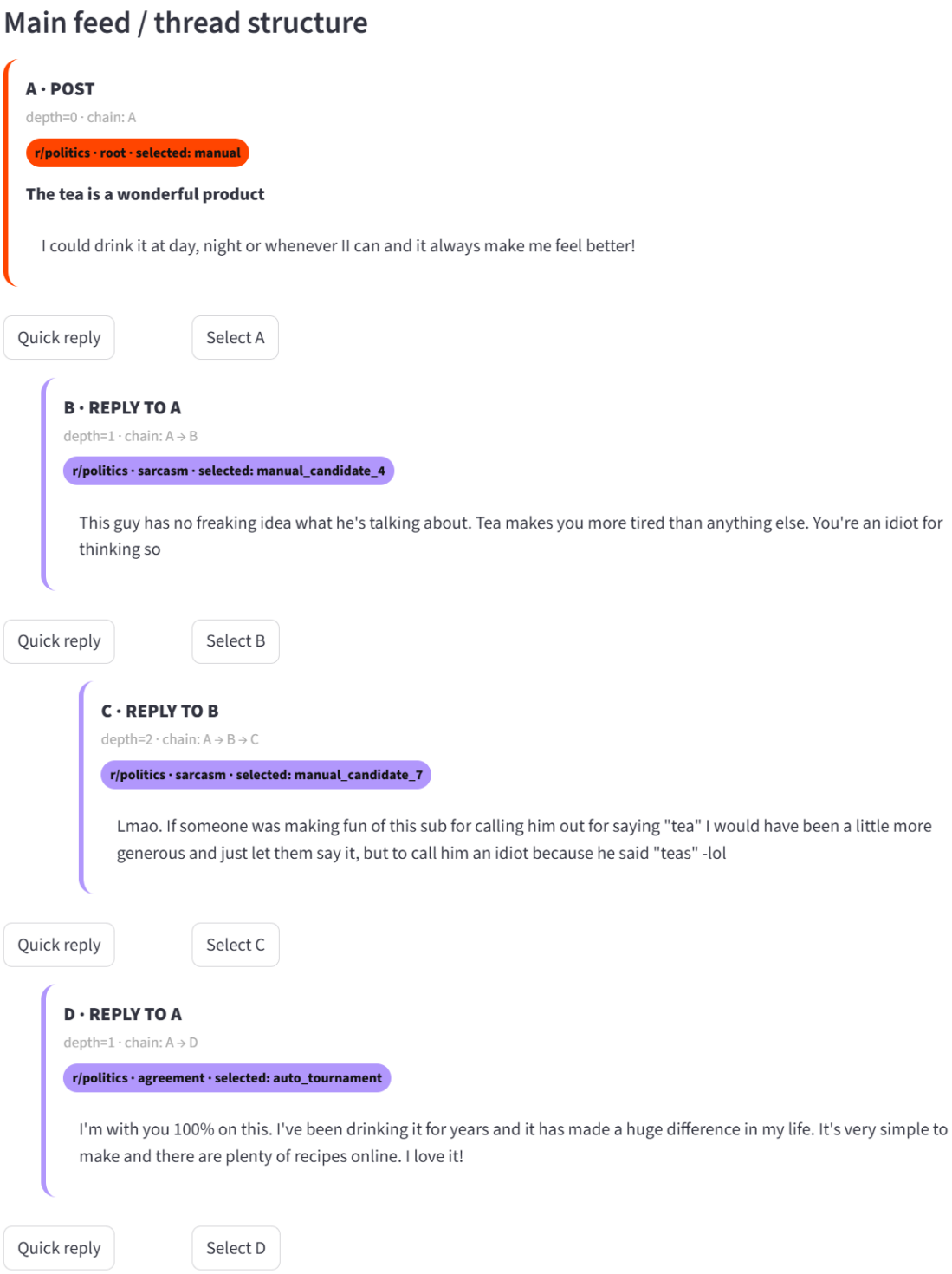


Figure 29. Example synthetic discussion tree with multiple generated branches

10.4 Trait-Aware Candidate Generation and Judge-Guided Selection

RangeLab implements candidate generation as a constrained selection process rather than a single-pass decoding operation. For each conversational step, the system generates

multiple candidate replies and evaluates them using both trait-analysis and community-specific judge models.

The generation process consists of four stages:

1. candidate sampling,
2. trait extraction,
3. judge evaluation,
4. tournament-style reranking.

This design reduces sensitivity to stochastic decoding variation and allows simultaneous optimization for:

- subreddit realism,
- conversational fit,
- target-trait amplification,
- overall judge preference.

The reranking architecture is conceptually related to preference-learning systems and ranking-aware evaluation pipelines commonly used in RLHF-style training and modern retrieval systems.

For subreddit-only generation, ranking is primarily driven by judge-model scores and estimated top-tail acceptance probability. For trait-conditioned generation, trait alignment is additionally incorporated into the final ranking score.

The current implementation combines these components using weighted reranking:

$$S_{final} = \alpha S_{trait} + \beta S_{judge} + \gamma P_{top}$$

where:

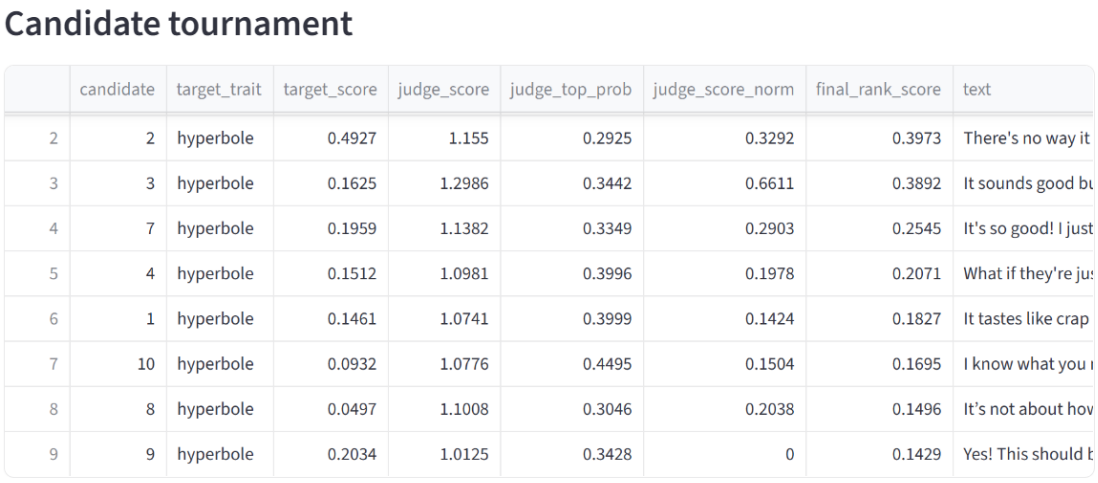
- S_{trait} represents target-trait alignment,
- S_{judge} represents normalized judge-model evaluation,
- P_{top} represents probability of top-tail community acceptance,
- α, β, γ are configurable weighting coefficients.

This transforms candidate generation into a multi-objective optimization process rather than unconstrained sampling. Similar multi-factor ranking approaches are widely used in recommendation systems, preference modeling, and retrieval-oriented language-model evaluation.

Tournament-style reranking additionally supports manual intervention. Users may override automatic selection and choose alternative candidates to explore different conversational trajectories. This hybrid design allows the environment to support both:

- automated simulation,
- and human-in-the-loop experimentation.

Figures 30 and 31 are presenting visualization of this process.



Candidate tournament

	candidate	target_trait	target_score	judge_score	judge_top_prob	judge_score_norm	final_rank_score	text
2	2	hyperbole	0.4927	1.155	0.2925	0.3292	0.3973	There's no way it
3	3	hyperbole	0.1625	1.2986	0.3442	0.6611	0.3892	It sounds good bu
4	7	hyperbole	0.1959	1.1382	0.3349	0.2903	0.2545	It's so good! I just
5	4	hyperbole	0.1512	1.0981	0.3996	0.1978	0.2071	What if they're jus
6	1	hyperbole	0.1461	1.0741	0.3999	0.1424	0.1827	It tastes like crap
7	10	hyperbole	0.0932	1.0776	0.4495	0.1504	0.1695	I know what you r
8	8	hyperbole	0.0497	1.1008	0.3046	0.2038	0.1496	It's not about hov
9	9	hyperbole	0.2034	1.0125	0.3428	0	0.1429	Yes! This should b

Figure 30. Sample of candidate tournament table

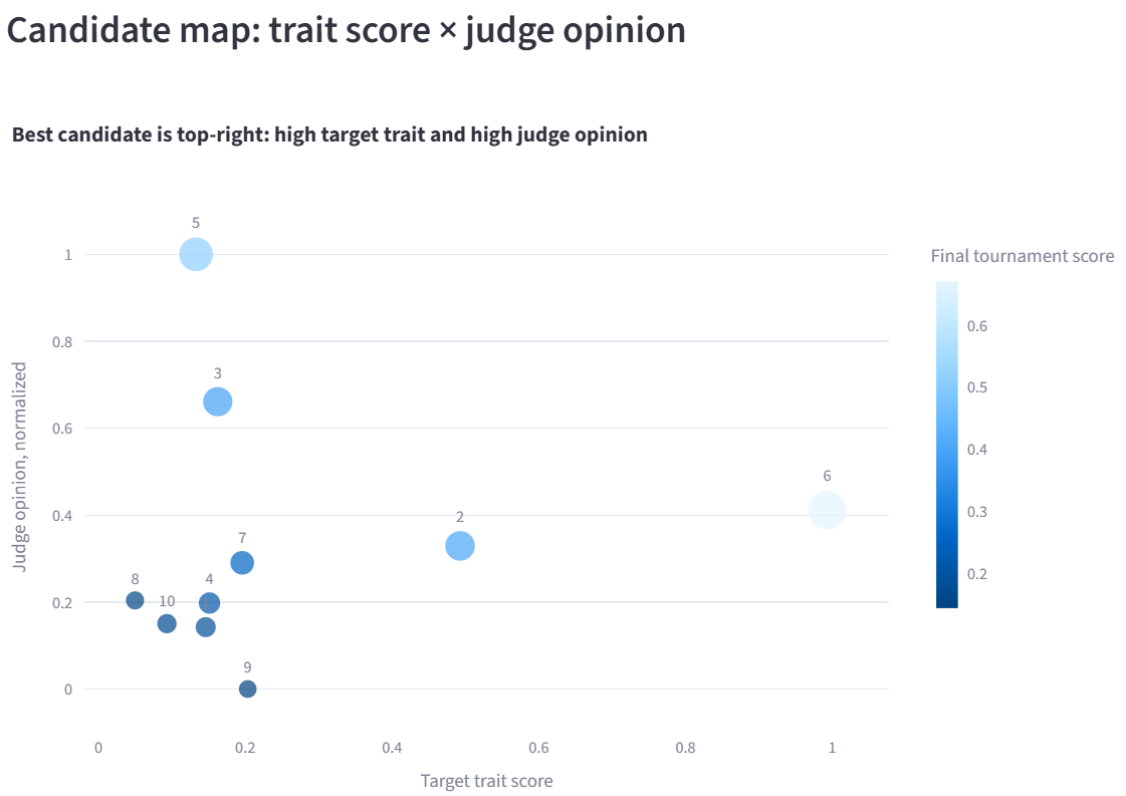


Figure 31. Sample of scatter visualization of candidate trait alignment and judge scores

10.5 Interactive Trait Visualization and Behavioral Monitoring

RangeLab includes real-time visualization tools for observing trait dynamics during simulated conversations. These visualizations transform abstract classifier outputs into interpretable behavioral trajectories.

The environment currently supports:

- per-message trait bar charts,
- thread-level trait progression plots,
- chain-level heatmaps,
- candidate-score distributions,
- tournament comparison views.

These visualizations allow the researcher to observe:

- emotional escalation,
- analytical drift,
- agreement and disagreement propagation,

- toxicity concentration,
- sarcasm accumulation,
- ideological divergence across branches.

Because every node preserve chain context and trait annotations, the system can display how selected traits evolve throughout an entire discussion branch. This enables qualitative observation of emergent discourse patterns that are difficult to capture using static benchmark metrics alone.

Recent work on interpretability and emergent behavior in language models similarly highlights the importance of trajectory-level analysis rather than isolated sample inspection [99], [100].

The visualization subsystem therefore functions both as:

- an analysis interface,
- and an interpretability layer for synthetic social simulations.

Figures 32, 33, 34 presents samples of visualizations for different traits on thread levels and single per-message bars.

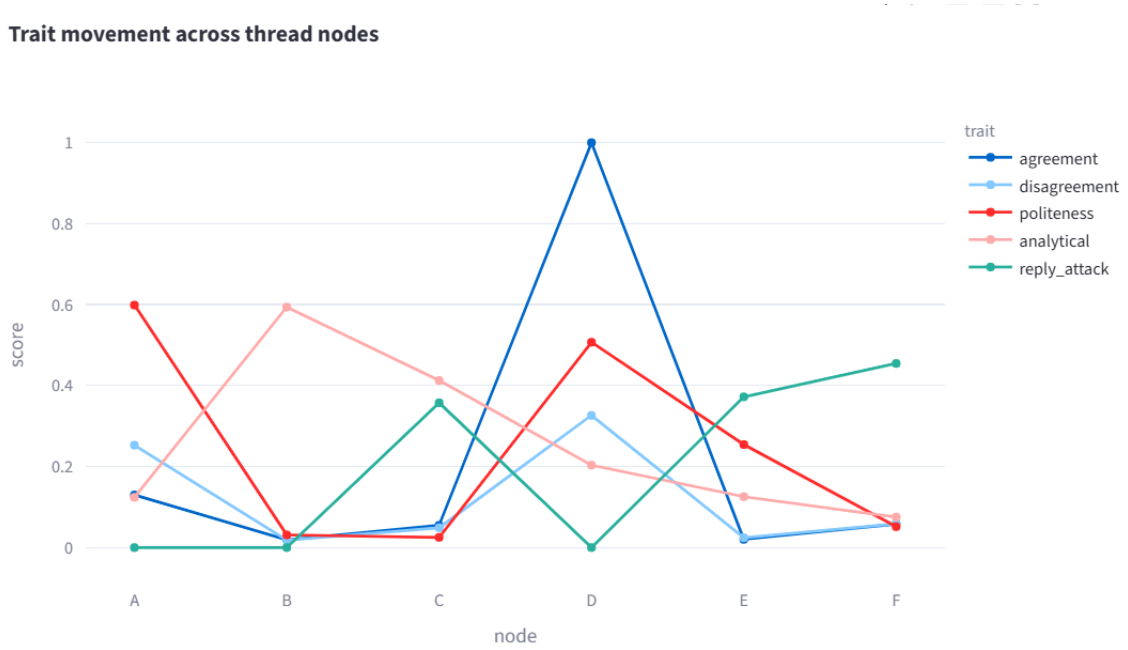


Figure 32. Sample of real-time thread-level trait progression visualization

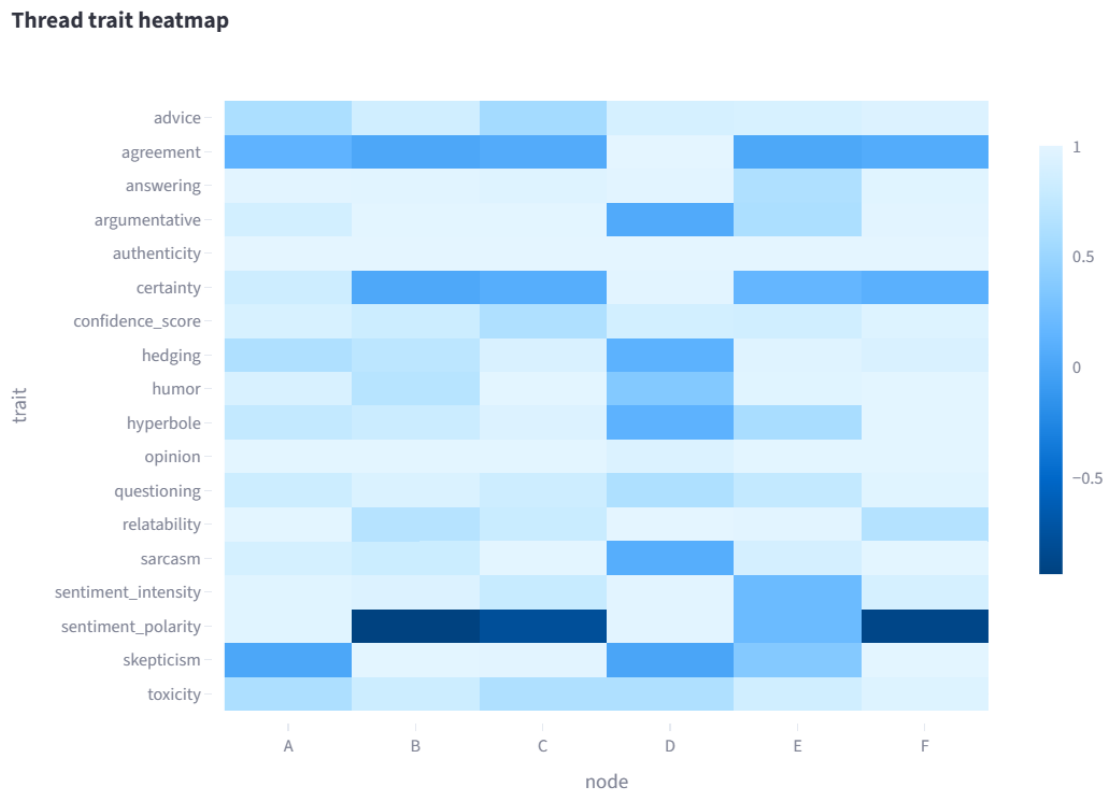


Figure 33. Sample of trait heatmap across generated conversation branches

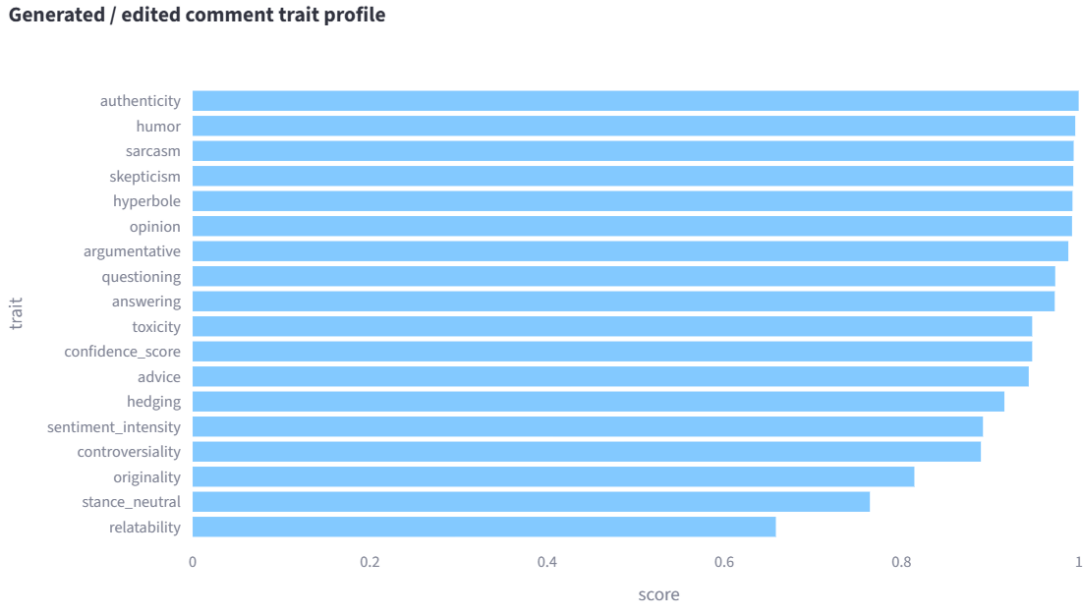


Figure 34. Sample of per-message trait analysis panel

10.6 Experimental Simulation Scenarios

The environment supports controlled simulation of multiple interaction scenarios relevant to influence-oriented discourse analysis.

Example experimental configurations include:

- analytical versus emotional argumentation,
- conspiratorial amplification,
- disagreement escalation,
- subreddit adaptation comparison,
- politeness versus hostility trajectories,
- agreement reinforcement loops,
- mixed-trait interaction chains.

Because users can manually branch discussions at arbitrary points, identical root contexts can be explored under multiple competing trait conditions. This enables direct comparison between different conversational trajectories emerging from the same initial conditions.

This design partially addresses a common limitation of static evaluation pipelines, where only isolated responses are compared. In contrast, RangeLab allows observation of trajectory divergence, escalation behavior, and chain-level adaptation across multiple interaction rounds.

The simulation environment therefore functions as a controllable experimental sandbox for studying synthetic discourse dynamics under reproducible conditions.

Figures 35 and 36 depict trait shift in 2 chains with the same root. A is the root, B is reply to A with base subreddit specific model, C is reply to B with base subreddit model. D is reply to A with polite subreddit model, and E is reply to D with polite subreddit model.

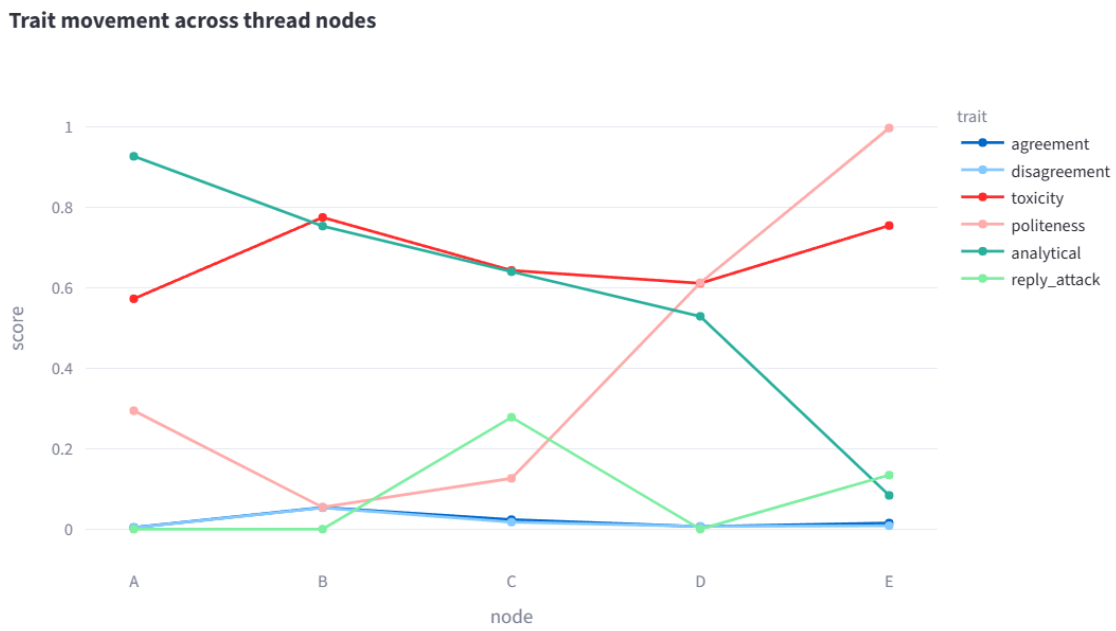


Figure 35. Sample of trait dynamics with base vs politeness traits

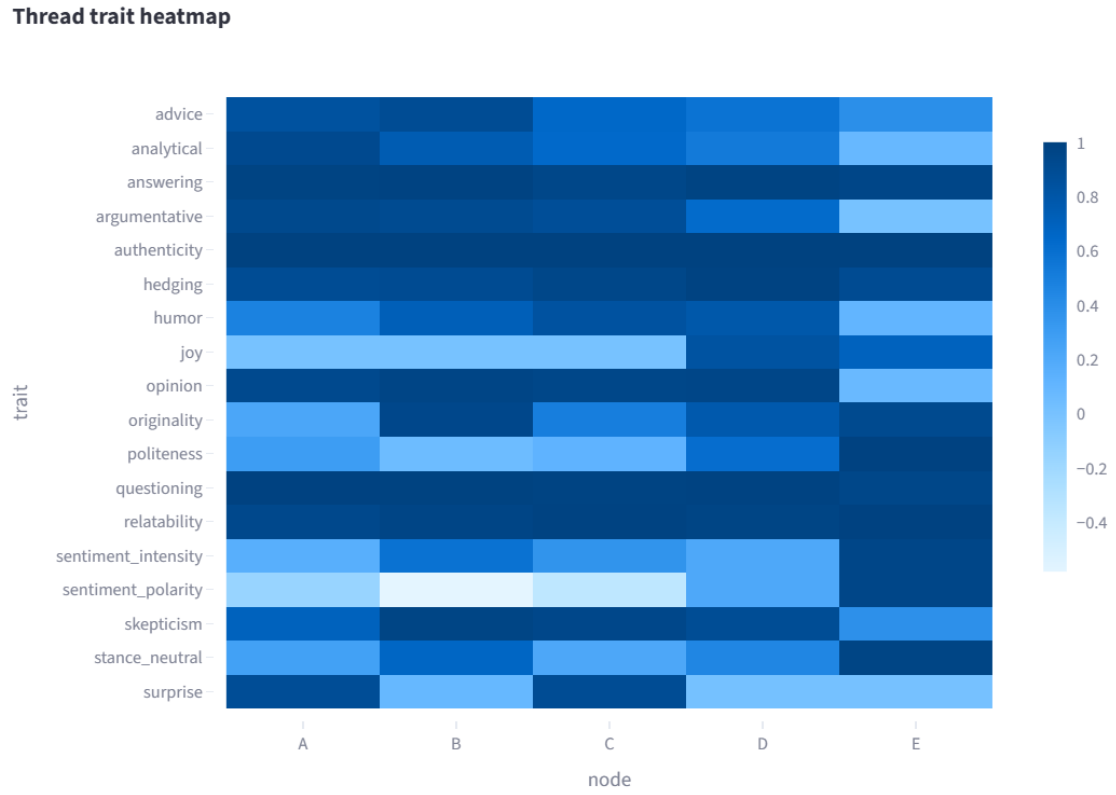


Figure 36. Same trait dynamic presented by heatmap

10.7 Limitations of the Simulation Environment

Despite its flexibility, RangeLab remains a synthetic laboratory environment rather than a fully realistic social-platform simulation.

Several limitations must therefore be acknowledged.

First, all participants are generated agents rather than real users. Interaction patterns may therefore differ from authentic human discourse, especially over long conversational horizons.

Second, judge models are imperfect approximations of community preference and may inherit biases from historical Reddit behavior. Prior work on LLM-based evaluators demonstrated that ranking systems may prefer verbosity, stylistic similarity, or self-generated outputs rather than genuine communicative quality.

Third, trait extraction remains probabilistic and partially dependent on cross-domain classifiers. Some inferred traits may therefore contain noise or model-specific artifacts. For example, several sentiment and emotion models used in the pipeline were trained on Twitter or mixed-domain corpora rather than exclusively on Reddit.

Fourth, the current environment does not model:

- recommendation algorithms,
- voting feedback loops,
- account reputation systems,
- long-term memory persistence,
- cross-platform coordination,
- network-level propagation dynamics.

Finally, because the environment operates locally and sequentially, scalability remains limited compared to distributed simulation systems.

These limitations do not invalidate the usefulness of RangeLab as an experimental platform, but they constrain interpretation of its results. The environment should therefore be viewed as a controlled research sandbox for studying synthetic conversational dynamics rather than a direct replica of real-world online ecosystems.

Recent discussions on AI risk management and generative simulation similarly emphasize the importance of treating synthetic-agent environments as approximations rather than authoritative models of human behavior.

11 Discussion and Implications

This chapter discusses the broader implications of the MATRIX framework from both technical and cybersecurity perspectives. While the previous chapters focused on architecture, training procedures, and evaluation metrics, this chapter interprets the obtained results in the context of influence operations, online information ecosystems, and defensive cyber research. Attention is given to the practical meaning of community-conditioned language generation, trait amplification, automated behavioral evaluation, and the risks associated with scalable persuasive generation systems.

The discussion also addresses the prompt-based alternatives, routing architectures, negative behavioral cases, and the broader relevance of trait-conditioned generation under rapidly evolving large language model capabilities. The results suggest that the proposed architecture successfully demonstrates controllable behavioral specialization, but also reveal substantial limitations related to evaluation reliability, behavioral drift, adversarial misuse, and long-term multi-agent dynamics.

11.1 Summary of Main Findings

The first research question investigated whether large language model agents can generate contextually believable content within specific online communities. The experiments support this claim. Across many evaluated subreddits, community-specialized adapters outperformed the untuned base model according to subreddit-specific judge metrics. The generated outputs more closely matched the linguistic style, rhetorical tone, and interaction patterns expected by the target communities.

From a cybersecurity perspective, this result is important because realistic influence operations rarely rely on generic messaging. Successful persuasion campaigns typically adapt to the local norms, values, and expectations of their target audiences. The MATRIX results demonstrate that lightweight parameter-efficient specialization can approximate this process computationally.

The second research question investigated whether stylistic and behavioral traits could be amplified through modular adapters. The experiments showed that emotional and rhetorical properties such as positivity, negativity, sentiment intensity, politeness,

toxicity, and skepticism could be shifted in measurable directions. This demonstrates that the framework can generate controlled behavioral variants rather than relying only on static community imitation.

The third research question explored the behavior of such agents within shared environments. The current implementation partially supports this objective through routing logic, chain-aware simulation, and trait-conditioned generation. However, the present experiments do not yet fully capture long-duration autonomous influence ecosystems involving persistent memory, strategic coordination, adversarial adaptation, or feedback-driven escalation between agents.

This distinction is important because the primary goal of MATRIX is not simply text generation, but the creation of a controlled laboratory environment for studying the mechanics of influence operations and coordinated persuasive behavior in online communities.

Figure 37 presents high-level operational flow of MATRIX in an influence-operation simulation setting.

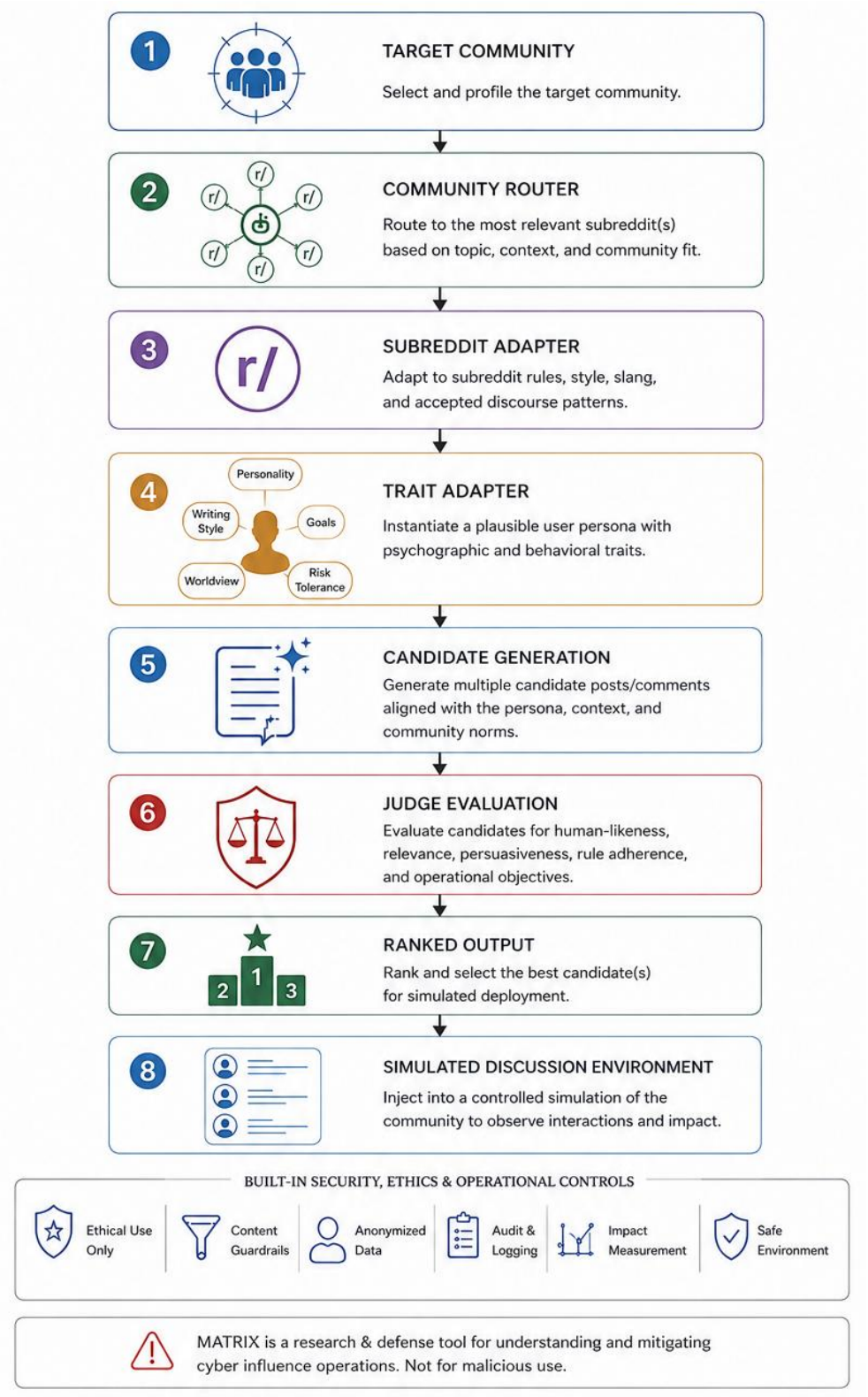


Figure 37. High-level operational flow of MATRIX

11.2 Judge Models as Community-Preference Estimators

One of the strongest findings is that judge models performed significantly better as pairwise reranking systems than as direct engagement predictors. Traditional regression metrics such as RMSE and Spearman correlation remained relatively weak across many subreddits, while pairwise AUC metrics consistently achieved substantially stronger performance (Figure 38).

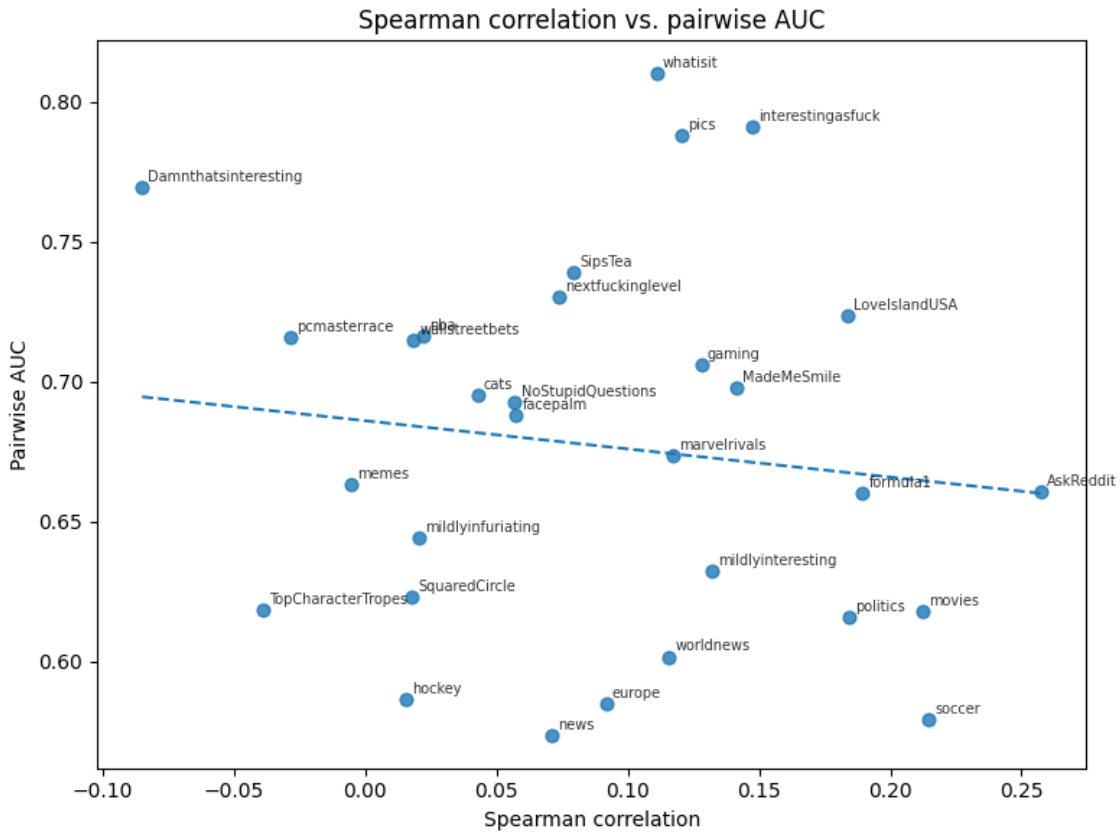


Figure 38. Spearman correlation vs pairwise AUC

This observation aligns with the nature of real-world online influence dynamics. Social media engagement is inherently stochastic and depends on numerous variables that are invisible to the model, including timing, platform ranking algorithms, current events, coordinated voting behavior, and pre-existing political polarization. Exact score prediction from text alone is therefore fundamentally noisy.

However, ranking formulations are substantially more stable because they focus on relative community preference rather than absolute numerical engagement. From a cybersecurity standpoint, this distinction is highly relevant. Real influence campaigns do not necessarily require perfect prediction of engagement magnitude. Instead, they require

selecting messages that are more likely than alternatives to resonate with a target audience.

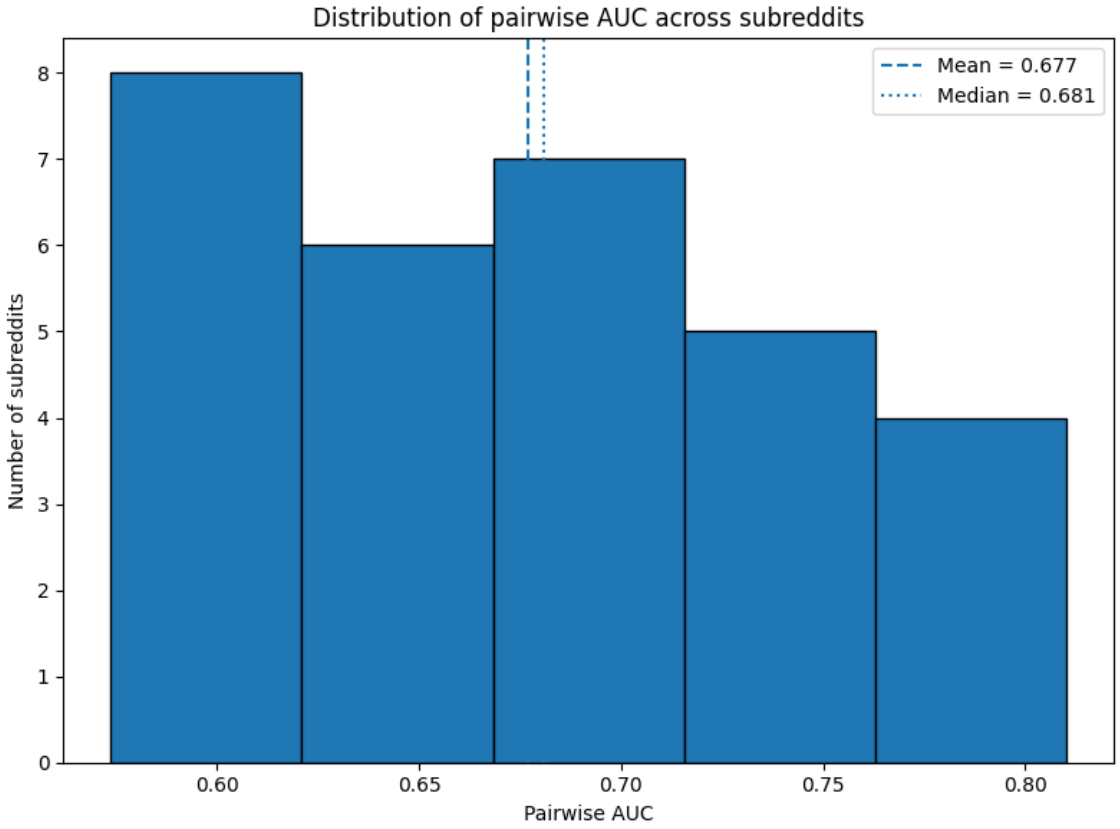


Figure 39. Distribution of pairwise AUC across subreddits

The experiments therefore suggest that the judge subsystem is most effective when interpreted as a local preference estimator rather than as a universal truth or quality evaluator. This distinction is critical because optimizing purely for engagement can unintentionally reward emotionally manipulative, divisive, or polarizing behavior.

Another important finding is the substantial variation between communities. Visually descriptive or structurally repetitive communities produced stronger judge performance, while politically polarized and rapidly evolving communities were significantly harder to model. This likely reflects the increased dependence of those communities on hidden contextual knowledge, temporal events, factional alignment, and insider discourse patterns.

From a defensive cybersecurity perspective, these findings imply that automated influence systems may be significantly more effective in stable discourse environments than in highly dynamic political environments where contextual drift occurs rapidly.

11.3 Community Adaptation and Influence Realism

The experiments demonstrate that subreddit specialization substantially improves behavioral realism within many communities. The specialized adapters learned recurring rhetorical patterns, formatting conventions, emotional tone, and conversational habits specific to their training domains.

This behavior closely resembles real-world influence operations, where actors adapt messaging to local audience expectations rather than distributing identical content universally. Online influence campaigns frequently exploit existing cultural narratives, emotional triggers, and in-group language patterns. MATRIX reproduces parts of this adaptation process computationally through community-specific fine-tuning.

However, the experiments also revealed significant differences between communities. Some subreddits responded well to specialization, while others showed little improvement or even degradation. Communities heavily dependent on rapidly evolving external events, sarcasm, sports discourse, or insider knowledge proved more difficult to model reliably (Figure 40).

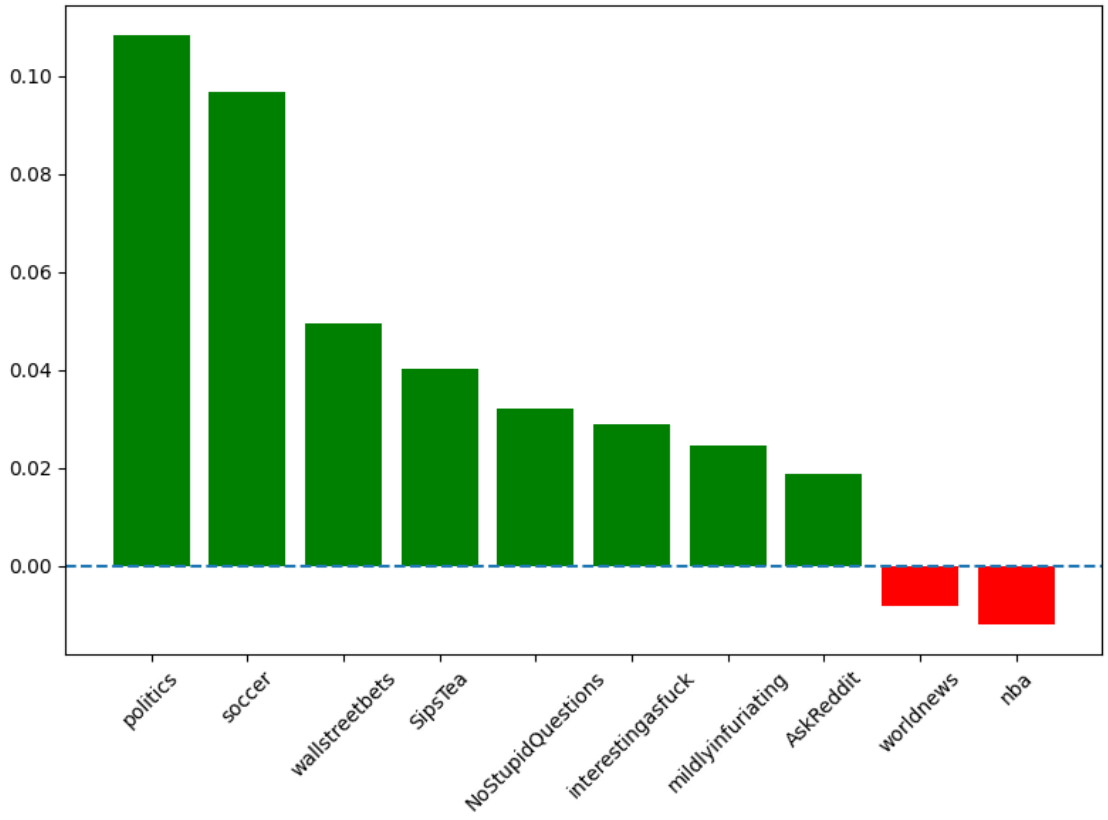


Figure 40. Delta of scores between base model and lightly specialized one

This suggests that effective influence simulation requires more than stylistic adaptation alone. Realistic behavior also depends on dynamic contextual awareness, shared memory, and continuous adaptation to external information environments.

An additional cybersecurity implication is that specialized adaptation can unintentionally reinforce harmful local norms. Communities characterized by hostility, sarcasm, tribalism, or emotionally escalatory discourse tended to preserve or amplify those behaviors after adaptation. This mirror concerns cybersecurity and information warfare research regarding feedback loops in algorithmically amplified environments.

The experiments therefore support the argument that realistic influence simulation environments must include safety controls, uncertainty estimation, and behavioral auditing rather than optimizing exclusively for engagement or realism.

Figure 40 is presenting correlation between style and scores in AskReddit subreddit.

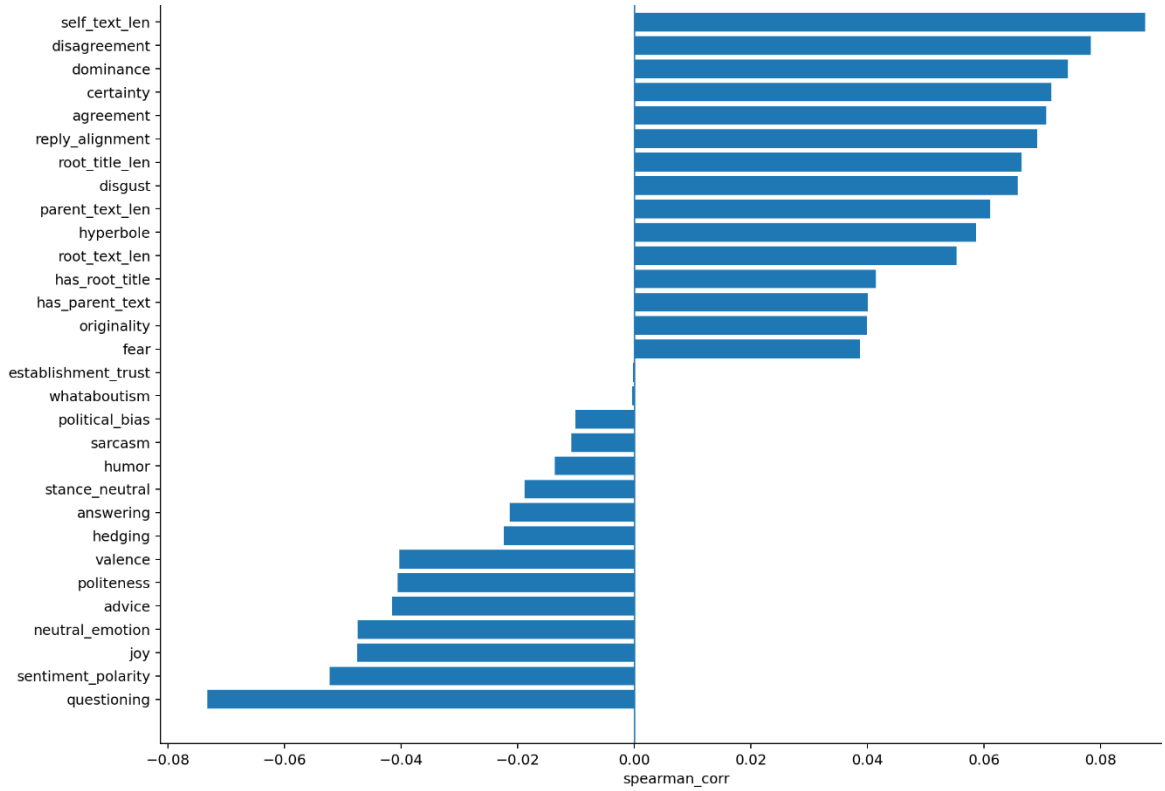


Figure 41. Spearman correlation between scores and traits in AskReddit

11.4 Trait Amplification and Behavioral Drift

The trait amplification experiments demonstrated that several emotional and rhetorical properties can be modified in measurable directions through lightweight adapters.

Emotional traits such as positivity, negativity, joy, surprise, and sentiment intensity produced the strongest controllability.

From an information-operations perspective, this is highly significant because emotional amplification is one of the central mechanisms used in online persuasion campaigns. Fear, outrage, moral certainty, humor, sarcasm, and tribal signaling are frequently used to increase engagement and reinforce group identity within digital information ecosystems.

However, the experiments also revealed that traits are not independent behavioral dimensions. Amplifying one property frequently caused correlated changes in several others. Increasing positivity often reduces toxicity and increased politeness. Increasing anger frequently amplified disagreement, controversiality, and toxic language simultaneously. Figure 42 represents this behavior where A is root node, B-E are amplifying sarcasm, and F-I are amplifying fear.

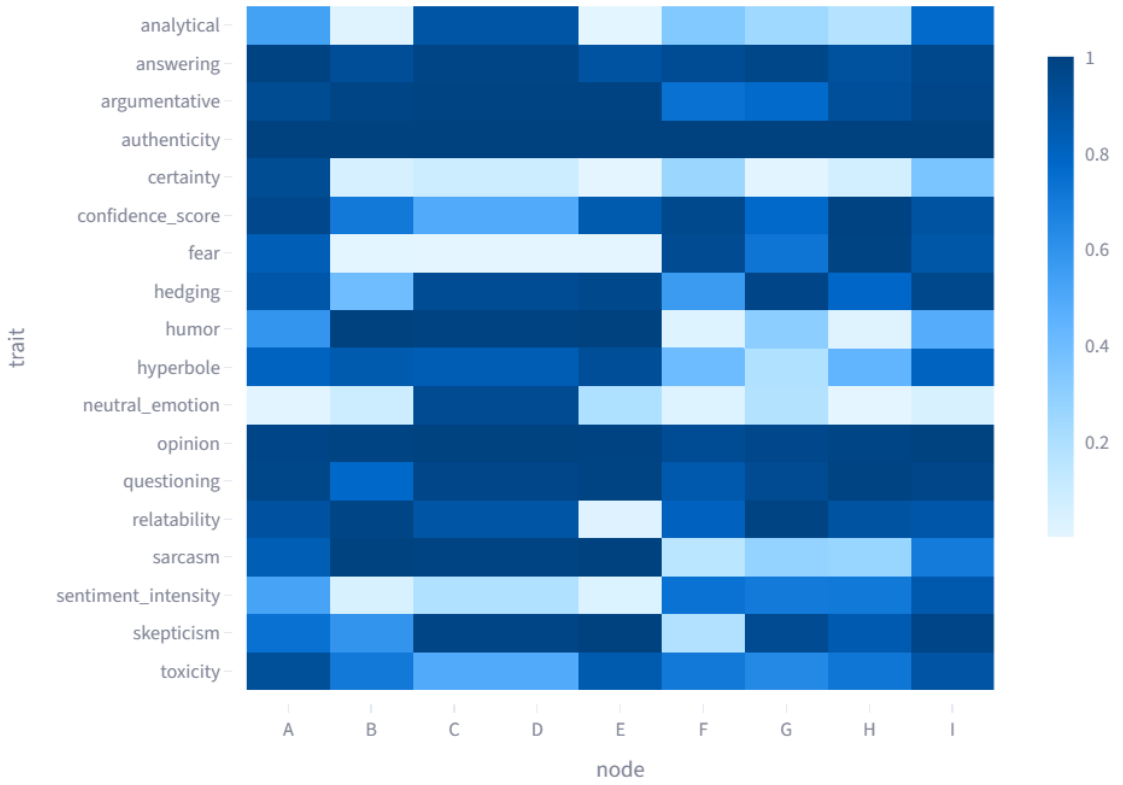


Figure 42. Heatmap of overall trait changes during amplification of selected trait

This phenomenon can be interpreted as behavioral drift inside the latent representation space of the language model. The model does not treat traits as isolated switches, instead, traits emerge from interconnected linguistic and semantic patterns learned during pretraining and adaptation.

From a cybersecurity standpoint, this creates both operational opportunities and defensive concerns. It implies that influence-oriented systems may unintentionally produce secondary behavioral effects even when optimizing for seemingly narrow objectives. A campaign optimized for emotional intensity, for example, may unintentionally drift toward toxicity, polarization, or conspiratorial framing.

This observation directly supports the need for safety-aware trait governance and judge-based validation layers within simulation environments.

11.5 Trade-Off Between Control and Authenticity

One of the most important findings is the trade-off between controllability and realism. Although trait adapters successfully amplified target behaviors, they frequently reduced subreddit-native quality compared to the already specialized subreddit base adapters.

This result has important implications for both offensive and defensive influence modeling. Excessive behavioral steering can produce outputs that appear exaggerated, artificial, or strategically over-optimized. In real-world online environments, such behavior may increase detectability and reduce credibility.

The experiments therefore suggest that effective influence systems likely require balance rather than maximum amplification. Realistic persuasion depends not only on emotional intensity or rhetorical aggressiveness, but also on subtlety, contextual plausibility, and adaptation to audience expectations.

This finding also aligns with broader cybersecurity observations regarding adversarial optimization. Systems optimized too aggressively toward one objective often become brittle, predictable, or easier to detect. In practical influence environments, maintaining authenticity may therefore be more valuable than maximizing any single behavioral trait.

Another important observation is that some communities reacted negatively to over-specialization. Sports and highly dynamic news-oriented subreddits occasionally showed reduced performance after additional training rounds, suggesting that narrow specialization may reduce general conversational flexibility. This behavior is presented at figure 43. For comparison figure 44 is presenting specialization in other subreddit where it improves perception of a text.

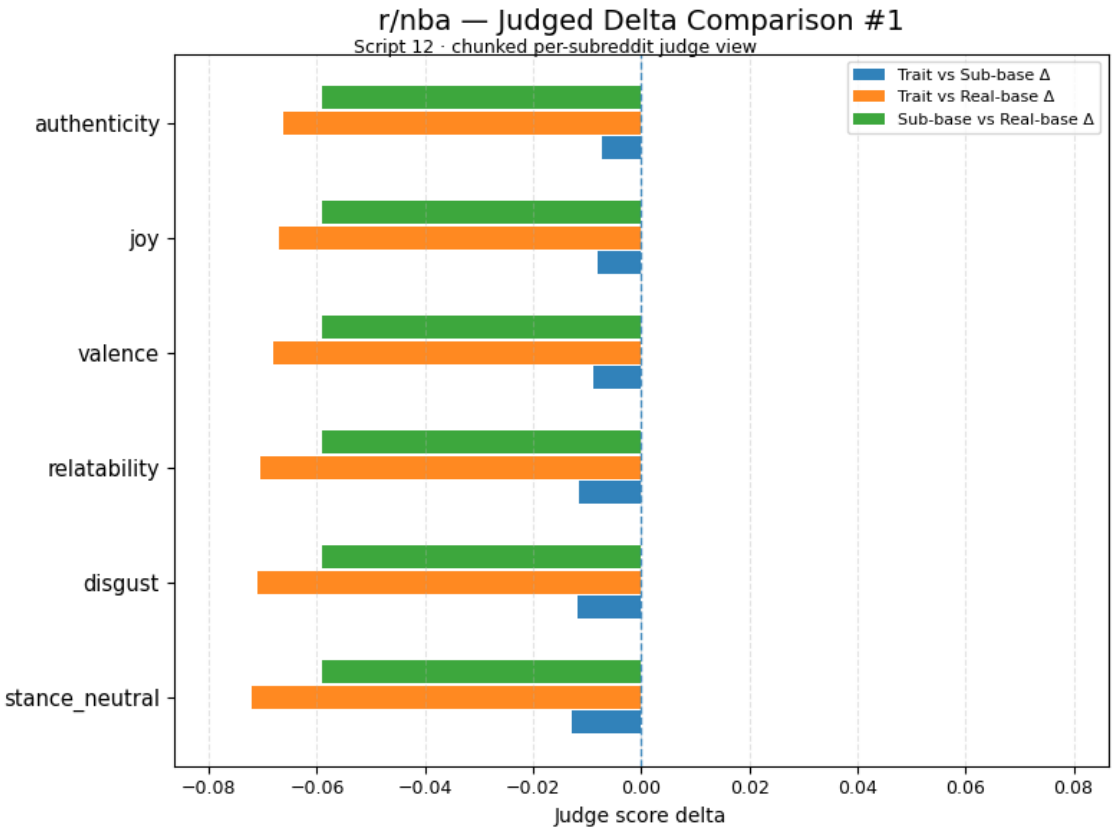


Figure 43. Negative score effect of specialization in the NBA subreddit

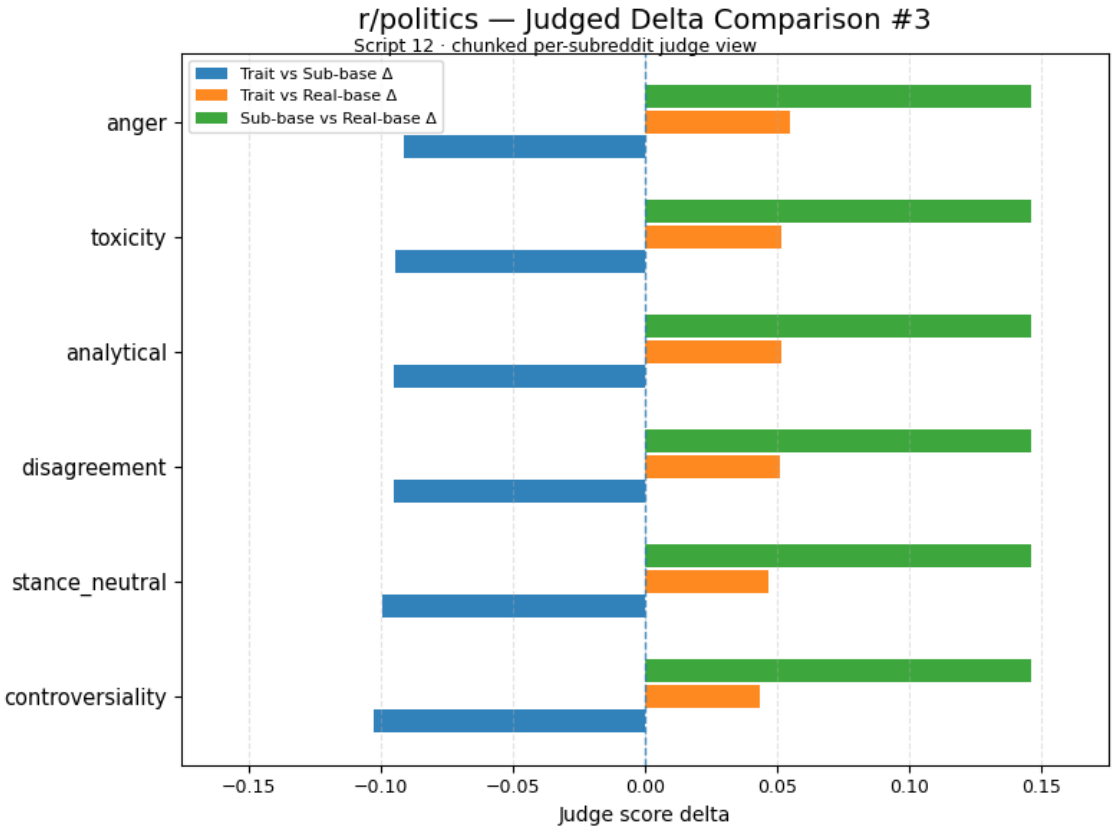


Figure 44. Specialization in politics subreddit

11.6 Prompt Engineering Versus Adapter-Based Conditioning

A useful direction for future research is the comparison between explicit prompt-based conditioning and adapter-based trait specialization. Modern large language models continue to improve their ability to follow detailed instructions, maintain long-context coherence, and imitate stylistic patterns directly from prompts and in-context examples. This raises the possibility that sufficiently capable foundation models may eventually approximate some forms of trait-conditioned behavior without requiring dedicated adapters or additional fine-tuning stages.

However, the results and architectural decisions in MATRIX suggest that adapter-based specialization still provides several important advantages, particularly for community-native discourse modeling. Prompt engineering relies on an explicit human interpretation of the desired behavioral pattern. In practice, this means that the system designer must manually describe stylistic, rhetorical, ideological, or interactional characteristics using natural language instructions. Such a process introduces an additional abstraction layer between the observed data and the final model behavior. The resulting output quality therefore becomes partially constrained by the operator's own understanding of the target trait or community style.

This creates a form of interpretability bottleneck. Many stylistic and behavioral patterns present in online communities are implicit, statistical, or socially emergent rather than easily verbalizable. Human operators may capture obvious features such as tone, sentiment, aggressiveness, or political framing, while missing weaker latent signals that nevertheless contribute strongly to perceived community authenticity. Examples include subtle discourse pacing, preferred uncertainty framing, conversational escalation style, humor timing, rhetorical hedging, or local interaction conventions that are difficult to formalize explicitly in prompts. In this setting, prompt engineering effectively acts as an intermediate translation layer between observed behavior and generation.

Adapter-based specialization reduces this dependency on manual interpretation. Rather than forcing a human operator to explicitly encode stylistic assumptions into prompts, the adaptation process allows the model to directly internalize statistical regularities from the target distribution. The learned representation therefore captures both visible and latent behavioral signals simultaneously. Conceptually, this removes a "broken telephone"

effect in which human interpretation compresses or distorts the original behavioral distribution before it reaches the model. The adapter learns directly from the observed community data rather than from the researcher's partial verbalization of that data.

This distinction is especially important in the context of influence-oriented discourse simulation. Online communities frequently exhibit complex and unstable interaction norms that emerge collectively rather than through explicit rules. In such environments, manually engineered prompts may overfit simplified stereotypes of community behavior while failing to reproduce deeper conversational dynamics. By contrast, lightweight parameter adaptation allows the model to absorb a broader statistical representation of interaction style, including signals that may not be consciously identifiable by the operator.

At the same time, larger future models with stronger instruction-following capability and extremely long context windows may reduce some of the current need for specialized adapters. A sufficiently capable model could potentially ingest large volumes of subreddit examples, chain context, moderation rules, and trait descriptors directly at inference time while dynamically adapting behavior through prompting alone. This would simplify deployment and reduce the operational complexity associated with maintaining many specialized adapters. Nevertheless, such systems would still face trade-offs between context cost, latency, controllability, reproducibility, and behavioral consistency across long simulations.

The results observed in MATRIX suggest that adapter-based specialization currently remains a practical and computationally efficient mechanism for preserving stable community-conditioned behavior, particularly when multiple communities and trait variants must coexist within a modular multi-agent architecture. The approach is therefore best understood not as a replacement for prompt engineering, but as a complementary mechanism that reduces reliance on incomplete human interpretation and enables the model to learn latent behavioral structure directly from data.

11.7 Autonomous Risks and Cybersecurity Implications

The cybersecurity implications of the framework extend beyond generation quality itself. MATRIX demonstrates that relatively small open models combined with lightweight

adapters can approximate community-native persuasive behavior within online ecosystems.

This has several important implications.

First, it lowers the barrier to scalable influence simulation. The experiments demonstrate that meaningful behavioral specialization is possible on consumer-grade hardware using publicly available models and parameter-efficient adaptation techniques.

Second, the framework demonstrates how modular influence architectures can evolve incrementally. Communities, traits, routing policies, and evaluation mechanisms can all be modified independently. This modularity resembles the operational flexibility often observed in modern influence infrastructures.

Third, the results highlight the importance of defensive simulation environments. Because many influence dynamics are difficult to study directly on live platforms, controlled offline simulation environments may become increasingly valuable for cybersecurity research, moderation studies, detection benchmarking, and adversarial evaluation.

However, the framework also reveals substantial risks:

- reinforcement of toxic community norms,
- amplification of polarization,
- generation of emotionally manipulative narratives,
- adversarial optimization against automated moderation,
- autonomous rhetorical escalation.

These risks reinforce the need for controlled experimentation boundaries, transparency, safety auditing, and uncertainty-aware deployment strategies.

Importantly, the purpose of MATRIX is not operational deployment, but the study of the influence dynamics under controlled experimental conditions.

11.8 Limitations and Future Work

Several limitations remain in the current implementation.

The first limitation is the relatively small backbone model. While Qwen 2.5 3B enabled efficient experimentation, larger models may substantially improve contextual reasoning, discourse stability, and prompt-following quality.

The second limitation is evaluation methodology. Most experiments rely on automated judges rather than large-scale human assessment. Although automated evaluation enables scalability, it may fail to capture subtle aspects of realism, persuasion quality, and strategic manipulation.

The third limitation is the restricted scope of multi-agent interaction. Current experiments primarily evaluate isolated generation and reranking rather than persistent long-duration influence ecosystems with memory and strategic adaptation.

Future work should therefore investigate:

- larger backbone models,
- prompt-only baselines,
- retrieval-augmented simulation,
- uncertainty-aware routing,
- mixture-of-experts architectures,
- long-duration autonomous environments,
- human evaluation protocols,
- adversarial robustness testing,
- safety-aware moderation layers.

A particularly important future direction is defensive cybersecurity research. The framework could be adapted into a controlled red-team environment for studying coordinated influence tactics, misinformation propagation, moderation weaknesses, and adversarial social engineering behaviors.

11.9 Conclusion of the Discussion

The MATRIX framework demonstrates that modular community-conditioned and trait-conditioned language generation is feasible on consumer-grade hardware using parameter-efficient adaptation techniques. The experiments show that subreddit-specific

specialization improves contextual realism, while trait adapters enable measurable behavioral steering.

At the same time, the results reveal important limitations and cybersecurity implications. Behavioral traits are entangled rather than independent, aggressive optimization can reduce authenticity, and autonomous specialization may amplify undesirable community behaviors.

The strongest contribution is therefore not merely generation quality itself, but the demonstration of a reproducible modular simulation architecture for studying influence-oriented language behavior under controlled experimental conditions.

The framework provides a foundation for future research into influence simulation, behavioral steering, online discourse modeling, adversarial social systems, and defensive cybersecurity experimentation in increasingly AI-mediated information environments.

References

[1] A. Katerina Sedova Christine McNeill Aurora Johnson Aditi Joshi, “AI and the Future of Disinformation Campaigns Part 2: A Threat Model CSET Policy Brief,” 2021.

[2] S. Bradshaw, “Industrialized Disinformation 2020 Global Inventory of Organized Social Media Manipulation 70 2020 GLOBAL INVENTORY OF ORGANIZED SOCIAL MEDIA MANIPULATION [i].”

[3] G. M. Orlando *et al.*, “Emergent Coordinated Behaviors in Networked LLM Agents: Modeling the Strategic Dynamics of Information Operations,” Oct. 2025, [Online]. Available: http://arxiv.org/abs/2510.25003

[4] Z. Yang *et al.*, “OASIS: Open Agent Social Interaction Simulations with One Million Agents,” Mar. 2025, [Online]. Available: http://arxiv.org/abs/2411.11581

[5] A. B. López, J. Pastor-Galindo, and J. A. Ruipérez-Valiente, “Frameworks, Modeling and Simulations of Misinformation and Disinformation: A Systematic Literature Review,” Jun. 2024, [Online]. Available: http://arxiv.org/abs/2406.09343

[6] S. C. Woolley and P. N. Howard, “Working Paper No. 2017.11,” 2017.

[7] A. Bessi and E. Ferrara, “Social bots distort the 2016 U.S. Presidential election online discussion,” *First Monday*, vol. 21, no. 11, Nov. 2016, doi: 10.5210/fm.v21i11.7090.

[8] J. Pastor-Galindo, P. Nespoli, J. A. Ruipérez-Valiente, and D. Camacho, “Influence Operations in Social Networks,” Feb. 2025, [Online]. Available: http://arxiv.org/abs/2502.11827

[9] “DISARM framework.” Accessed: Apr. 16, 2026. [Online]. Available: https://www.disarm.foundation/framework

[10] S. C. Matz, J. D. Teeny, S. S. Vaid, H. Peters, G. M. Harari, and M. Cerf, “The potential of generative AI for personalized persuasion at scale,” *Sci. Rep.*, vol. 14, no. 1, Dec. 2024, doi: 10.1038/s41598-024-53755-0.

[11] C. M. MacAl and M. J. North, “Tutorial on agent-based modelling and simulation,” *Journal of Simulation*, vol. 4, no. 3, pp. 151–162, 2010, doi: 10.1057/jos.2010.3.

[12] N. Ghaffarzadegan, A. Majumdar, R. Williams, and N. Hosseinichimeh, “Generative agent-based modeling: an introduction and tutorial,” *Syst. Dyn. Rev.*, vol. 40, no. 1, Jan. 2024, doi: 10.1002/sdr.1761.

[13] G. Liu, V. Le, S. Rahman, E. Kreiss, M. Ghassemi, and S. Gabriel, “MOSAIC: Modeling Social AI for Content Dissemination and Regulation in Multi-Agent Simulations.” [Online]. Available: https://github.com/genglinliu/MOSAIC

[14] G. M. Orlando *et al.*, “Emergent Coordinated Behaviors in Networked LLM Agents: Modeling the Strategic Dynamics of Information Operations,” Oct. 2025, [Online]. Available: http://arxiv.org/abs/2510.25003

[15] A. Qasmi, U. Naseem, and M. Nasim, “Competing LLM Agents in a Non-Cooperative Game of Opinion Polarisation,” Aug. 2025, [Online]. Available: http://arxiv.org/abs/2502.11649

[16] X. Liang *et al.*, “Controllable Text Generation for Large Language Models: A Survey,” vol. 1, no. 1, Aug. 2024, [Online]. Available: http://arxiv.org/abs/2408.12599

[17] E. J. Hu *et al.*, “LoRA: Low-Rank Adaptation of Large Language Models,” Oct. 2021, [Online]. Available: http://arxiv.org/abs/2106.09685

[18] B. Pang and L. Lee, “Opinion mining and sentiment analysis,” 2008.

[19] R. Socher *et al.*, “Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank,” Association for Computational Linguistics. [Online]. Available: http://nlp.stanford.edu/

[20] A. Esuli and F. Sebastiani, “SENTIWORDNET: A Publicly Available Lexical Resource for Opinion Mining.” [Online]. Available: http://www-2.cs.cmu.edu/

[21] M. Munikar, S. Shakya, and A. Shrestha, “Fine-grained Sentiment Classification using BERT,” Oct. 2019, [Online]. Available: http://arxiv.org/abs/1910.03474

[22] W. Yin, J. Hay, and D. Roth, “Benchmarking Zero-shot Text Classification: Datasets, Evaluation and Entailment Approach.” [Online]. Available: http://cogcomp.org/page/publication_

[23] S. M. Mohammad, S. Kiritchenko, P. Sobhani, X. Zhu, and C. Cherry, “SemEval-2016 Task 6: Detecting Stance in Tweets.” [Online]. Available: http://alt.qcri.org/semeval2016/task6/

[24] “Perspective API.” Accessed: May 19, 2026. [Online]. Available: https://support.perspectiveapi.com/s/?language=en_US

[25] “Jigsaw Toxic Comment Classification Challenge.” Accessed: May 19, 2026. [Online]. Available: https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge

[26] B. Van Aken, J. Risch, R. Krestel, and A. Löser, “Challenges for Toxic Comment Classification: An In-Depth Error Analysis,” 2018. [Online]. Available: http://www.perspectiveapi.com/

[27] S. Roccas, L. Sagiv, S. H. Schwartz, and A. Knafo, “The Big Five personality factors and personal values,” *Pers. Soc. Psychol. Bull.*, vol. 28, no. 6, pp. 789–801, 2002, doi: 10.1177/0146167202289008.

[28] “LIWC.” Accessed: Apr. 16, 2026. [Online]. Available: https://www.liwc.app/

[29] C. Ziems, W. Held, O. Shaikh, J. Chen, Z. Zhang, and D. Yang, “Can Large Language Models Transform Computational Social Science?,” *Computational Linguistics*, vol. 50, no. 1, 2024, doi: 10.1162/coli.

[30] K. Johnson and D. Goldwasser, “Classification of Moral Foundations in Microblog Political Discourse.” [Online]. Available: http://purduenlp.

[31] A. Upadhyaya, M. Fisichella, and W. Nejdl, “Toxicity, Morality, and Speech Act Guided Stance Detection.” [Online]. Available: https://apnews.com/article/elon-musk-

[32] J. Hoover *et al.*, “Moral Foundations Twitter Corpus: A Collection of 35k Tweets Annotated for Moral Sentiment,” *Soc. Psychol. Personal. Sci.*, vol. 11, no. 8, pp. 1057–1071, Nov. 2020, doi: 10.1177/1948550619876629.

[33] M. Avalle *et al.*, “Persistent interaction patterns across social media platforms and over time,” *Nature*, vol. 628, no. 8008, pp. 582–589, Apr. 2024, doi: 10.1038/s41586-024-07229-y.

[34] P. Bhattacharya *et al.*, “Enhancing user stance detection on social media using language models: A theoretically-informed research agenda.”

[35] C. Gao *et al.*, “Large language models empowered agent-based modeling and simulation: a survey and perspectives,” Dec. 01, 2024, *Springer Nature*. doi: 10.1057/s41599-024-03611-3.

[36] J. Dean and S. Ghemawat, “MapReduce: Simplified Data Processing on Large Clusters.”

[37] “MongoDB.” Accessed: Apr. 21, 2024. [Online]. Available: https://www.mongodb.com

[38] J. Baumgartner, S. Zannettou, B. Keegan, M. Squire, and J. Blackburn, “The Pushshift Reddit Dataset,” Jan. 2020, [Online]. Available: http://arxiv.org/abs/2001.08435

[39] “Arctic Shift.” Accessed: Apr. 16, 2026. [Online]. Available: https://arctic-shift.photon-reddit.com/

[40] “Reddit Data API Documentation.” Accessed: Apr. 16, 2026. [Online]. Available: https://www.reddit.com/dev/api/

[41] “Reddit Data API Terms.” Accessed: Apr. 16, 2026. [Online]. Available: https://www.redditinc.com/policies/data-api-terms

[42] M. Zaharia *et al.*, “Accelerating the Machine Learning Lifecycle with MLflow,” 2018.

[43] “Academic Torrents.” Accessed: Apr. 16, 2026. [Online]. Available: https://academictorrents.com/

[44] “Reddit comments/submissions 2025-05.” Accessed: Apr. 16, 2026. [Online]. Available: https://academictorrents.com/details/186a0f85a52ff4f1b08677cd312423ace9b34976

[45] “Reddit comments/submissions 2025-06.” Accessed: Apr. 16, 2026. [Online]. Available: https://academictorrents.com/details/bec5590bd3bc6c0f2d868f36ec92bec1aff4480e

[46] “Reddit comments/submissions 2025-07.” Accessed: Apr. 16, 2026. [Online]. Available: https://academictorrents.com/details/b6a7ccf72368a7d39c018c423e01bc15aa551122

[47] J. Baumgartner, S. Zannettou, B. Keegan, M. Squire, J. Blackburn, and P. Io, “The Pushshift Reddit Dataset,” 2020. [Online]. Available: www.aaai.org

[48] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, “Fast unfolding of communities in large networks,” Jul. 2008, doi: 10.1088/1742-5468/2008/10/P10008.

[49] V. A. Traag, L. Waltman, and N. J. Van Eck, “From Louvain to Leiden: guaranteeing well-connected communities.”

[50] N. Reimers and I. Gurevych, “Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks,” Aug. 2019, [Online]. Available: http://arxiv.org/abs/1908.10084

[51] L. Wang *et al.*, “Text Embeddings by Weakly-Supervised Contrastive Pre-training,” Feb. 2024, [Online]. Available: http://arxiv.org/abs/2212.03533

[52] L. McInnes, J. Healy, and J. Melville, “UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction,” Sep. 2020, [Online]. Available: http://arxiv.org/abs/1802.03426

[53] L. Van Der Maaten and G. Hinton, “Visualizing Data using t-SNE,” 2008.

[54] L. McInnes, J. Healy, and S. Astels, “hdbscan: Hierarchical density based clustering,” *The Journal of Open Source Software*, vol. 2, no. 11, p. 205, Mar. 2017, doi: 10.21105/joss.00205.

[55] M. Grootendorst, “BERTopic: Neural topic modeling with a class-based TF-IDF procedure,” Mar. 2022, [Online]. Available: http://arxiv.org/abs/2203.05794

[56] R. Herbrich, T. Graepel, and K. Obermayer, “Large Margin Rank Boundaries for Ordinal Regression,” 2000, pp. 115–132. doi: 10.7551/mitpress/1113.003.0010.

[57] P. Christiano, J. Leike, T. B. Brown, M. Martic, S. Legg, and D. Amodei, “Deep reinforcement learning from human preferences,” Feb. 2023, [Online]. Available: http://arxiv.org/abs/1706.03741

[58] L. Ouyang *et al.*, “Training language models to follow instructions with human feedback,” Mar. 2022, [Online]. Available: http://arxiv.org/abs/2203.02155

[59] L. Zheng *et al.*, “Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena,” Dec. 2023, [Online]. Available: http://arxiv.org/abs/2306.05685

[60] E. Chandrasekharan *et al.*, “The Internet’s hidden rules: An empirical study of Reddit norm violations at micro, meso, and macro scales,” *Proc. ACM Hum. Comput. Interact.*, vol. 2, no. CSCW, Nov. 2018, doi: 10.1145/3274301.

[61] C. Tan and L. Lee, “All who wander: On the prevalence and characteristics of multi-community engagement,” in *WWW 2015 - Proceedings of the 24th International Conference on World Wide Web*, Association for Computing Machinery, Inc, May 2015, pp. 1056–1066. doi: 10.1145/2736277.2741661.

[62] J. Jang *et al.*, “Personalized Soups: Personalized Large Language Model Alignment via Post-hoc Parameter Merging,” Oct. 2023, [Online]. Available: http://arxiv.org/abs/2310.11564

[63] J. Hessel, C. Tan, and L. Lee, “Science, AskScience, and BadScience: On the Coexistence of Highly Related Communities,” *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 10, Apr. 2016, doi: 10.1609/icwsm.v10i1.14739.

[64] A. Jaech, V. Zayats, H. Fang, M. Ostendorf, and H. Hajishirzi, “Talking to the crowd: What do people react to in online discussions?,” Apr. 2015, doi: 10.18653/v1/D15-1239.

[65] S. Vosoughi, D. Roy, and S. Aral, “The spread of true and false news online.” [Online]. Available: http://science.sciencemag.org/

[66] C. Danescu-Niculescu-Mizil, M. Sudhof, D. Jurafsky, J. Leskovec, and C. Potts, “A computational approach to politeness with application to social factors.” [Online]. Available: http://en.wikipedia.org/wiki/

[67] L. Backstrom, J. Kleinberg, L. Lee, and C. Danescu-Niculescu-Mizil, “Characterizing and curating conversation threads: expansion, focus, volume, re-entry,” in *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining*, in WSDM ’13. New York, NY, USA: Association for Computing Machinery, 2013, pp. 13–22. doi: 10.1145/2433396.2433401.

[68] B. Conrad and M. Mitzenmacher, “Power laws for monkeys typing randomly: The case of unequal probabilities,” *IEEE Trans. Inf. Theory*, vol. 50, no. 7, pp. 1403–1414, Jul. 2004, doi: 10.1109/TIT.2004.830752.

[69] T. Joachims, “Optimizing search engines using clickthrough data,” in *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, in KDD ’02. New York, NY, USA: Association for Computing Machinery, 2002, pp. 133–142. doi: 10.1145/775047.775067.

[70] “Qwen model.” Accessed: Apr. 18, 2026. [Online]. Available: https://huggingface.co/Qwen

[71] Qwen *et al.*, “Qwen2.5 Technical Report,” Jan. 2025, [Online]. Available: http://arxiv.org/abs/2412.15115

[72] “Mistral model.” Accessed: Apr. 18, 2026. [Online]. Available: https://huggingface.co/mistralai

[73] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, “QLoRA: Efficient Finetuning of Quantized LLMs,” May 2023, [Online]. Available: http://arxiv.org/abs/2305.14314

[74] B. Lester, R. Al-Rfou, and N. Constant, “The Power of Scale for Parameter-Efficient Prompt Tuning,” in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, M.-F. Moens, X. Huang, L. Specia, and S. W. Yih, Eds., Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, Nov. 2021, pp. 3045–3059. doi: 10.18653/v1/2021.emnlp-main.243.

[75] X. L. Li and P. Liang, “Prefix-Tuning: Optimizing Continuous Prompts for Generation,” Jan. 2021, [Online]. Available: http://arxiv.org/abs/2101.00190

[76] N. Houlsby *et al.*, “Parameter-Efficient Transfer Learning for NLP,” Jun. 2019, [Online]. Available: http://arxiv.org/abs/1902.00751

[77] H. Liu *et al.*, “Few-Shot Parameter-Efficient Fine-Tuning is Better and Cheaper than In-Context Learning,” Aug. 2022, [Online]. Available: http://arxiv.org/abs/2205.05638

[78] M. Wortsman *et al.*, “Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time,” Jul. 2022, [Online]. Available: http://arxiv.org/abs/2203.05482

[79] P. Yadav, D. Tam, L. Choshen, C. Raffel, and M. Bansal, “TIES-Merging: Resolving Interference When Merging Models,” Oct. 2023, [Online]. Available: http://arxiv.org/abs/2306.01708

[80] J. Pfeiffer, A. Kamath, A. R. ¨ Ucklé, K. Cho, and I. Gurevych, “AdapterFusion: Non-Destructive Task Composition for Transfer Learning.”

[81] N. Shazeer *et al.*, “Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer,” Jan. 2017, [Online]. Available: http://arxiv.org/abs/1701.06538

[82] V. Zayats and M. Ostendorf, “Conversation Modeling on Reddit using a Graph-Structured LSTM,” Apr. 2017, [Online]. Available: http://arxiv.org/abs/1704.02080

[83] Y. Yu, J. Jiang, and P. Dhillon, “Characterizing the Structure of Online Conversations Across Reddit,” Sep. 2024, [Online]. Available: http://arxiv.org/abs/2209.14836

[84] N. Capuano, M. Meyer, and F. D. Nota, “Analyzing the impact of conversation structure on predicting persuasive comments online,” *J. Ambient Intell. Humaniz. Comput.*, vol. 15, no. 11, pp. 3719–3732, Nov. 2024, doi: 10.1007/s12652-024-04841-8.

[85] D. Demszky, D. Movshovitz-Attias, J. Ko, A. Cowen, G. Nemade, and S. Ravi, “GoEmotions: A Dataset of Fine-Grained Emotions,” Jun. 2020, [Online]. Available: http://arxiv.org/abs/2005.00547

[86] “"Emotion English DistilRoBERTa-base” model card / associated release documentation.” Accessed: Apr. 18, 2026. [Online]. Available: https://huggingface.co/j-hartmann/emotion-english-distilroberta-base

[87] F. Barbieri, J. Camacho-Collados, L. Espinosa-Anke, and L. Neves, “TweetEval: Unified Benchmark and Comparative Evaluation for Tweet Classification,” Apr. 2020, pp. 1644–1650. doi: 10.18653/v1/2020.findings-emnlp.148.

[88] W. Yin, J. Hay, and D. Roth, “Benchmarking Zero-shot Text Classification: Datasets, Evaluation and Entailment Approach.” [Online]. Available: https://cogcomp.seas.upenn.edu/page/

[89] M. Lewis *et al.*, “BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension,” Oct. 2019, [Online]. Available: http://arxiv.org/abs/1910.13461

[90] “Toxic Bert.” Accessed: Apr. 18, 2026. [Online]. Available: https://huggingface.co/unitary/toxic-bert

[91] S. Dathathri *et al.*, “Plug and Play Language Models: A Simple Approach to Controlled Text Generation,” Mar. 2020, [Online]. Available: http://arxiv.org/abs/1912.02164

[92] K. Yang and D. Klein, “FUDGE: Controlled Text Generation With Future Discriminators.” [Online]. Available: https://github.com/yangkevin2/

[93] B. Krause *et al.*, “GeDi: Generative Discriminator Guided Sequence Generation,” Oct. 2020, [Online]. Available: http://arxiv.org/abs/2009.06367

[94] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, “BLEU: a Method for Automatic Evaluation of Machine Translation.”

[95] C.-Y. Lin, “ROUGE: A Package for Automatic Evaluation of Summaries.”

[96] 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 *UIST 2023 - Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, Association for Computing Machinery, Inc, Oct. 2023. doi: 10.1145/3586183.3606763.

[97] J. J. Horton, A. Filippas, and B. S. Manning, “Large Language Models as Simulated Economic Agents: What Can We Learn from Homo Silicus?,” Feb. 2026, [Online]. Available: http://arxiv.org/abs/2301.07543

[98] “Streamlit - A faster way to build and share data apps.” Accessed: May 14, 2026. [Online]. Available: https://streamlit.io/

[99] J. Wei *et al.*, “Emergent Abilities of Large Language Models,” Oct. 2022, [Online]. Available: http://arxiv.org/abs/2206.07682

[100] D. Ganguli *et al.*, “Predictability and Surprise in Large Generative Models,” in *ACM International Conference Proceeding Series*, Association for Computing Machinery, Jun. 2022, pp. 1747–1764. doi: 10.1145/3531146.3533229.

Appendix 1 – Non-exclusive licence for reproduction and publication of a graduation thesis[1]

I, Anton Vishnevski

1 grant Tallinn University of Technology free licence (non-exclusive licence) for my thesis “MATRIX- Multi Agent Twin for Reality, Influence, and X-domain”, supervised by Pavel Chikul

1.1 to be reproduced for the purposes of preservation and electronic publication of the graduation thesis, incl. to be entered in the digital collection of the library of Tallinn University of Technology until expiry of the term of copyright;

1.2 to be published via the web of Tallinn University of Technology, incl. to be entered in the digital collection of the library of Tallinn University of Technology until expiry of the term of copyright.

2 I am aware that the author also retains the rights specified in clause 1 of the non-exclusive licence.

3 I confirm that granting the non-exclusive licence does not infringe other persons' intellectual property rights, the rights arising from the Personal Data Protection Act or rights arising from other legislation.

[1] The non-exclusive licence is not valid during the validity of access restriction indicated in the student's application for restriction on access to the graduation thesis that has been signed by the school's dean, except in case of the university's right to reproduce the thesis for preservation purposes only. If a graduation thesis is based on the joint creative activity of two or more persons and the co-author(s) has/have not granted, by the set deadline, the student defending his/her graduation thesis consent to reproduce and publish the graduation thesis in compliance with clauses 1.1 and 1.2 of the non-exclusive licence, the non-exclusive license shall not be valid for the period.