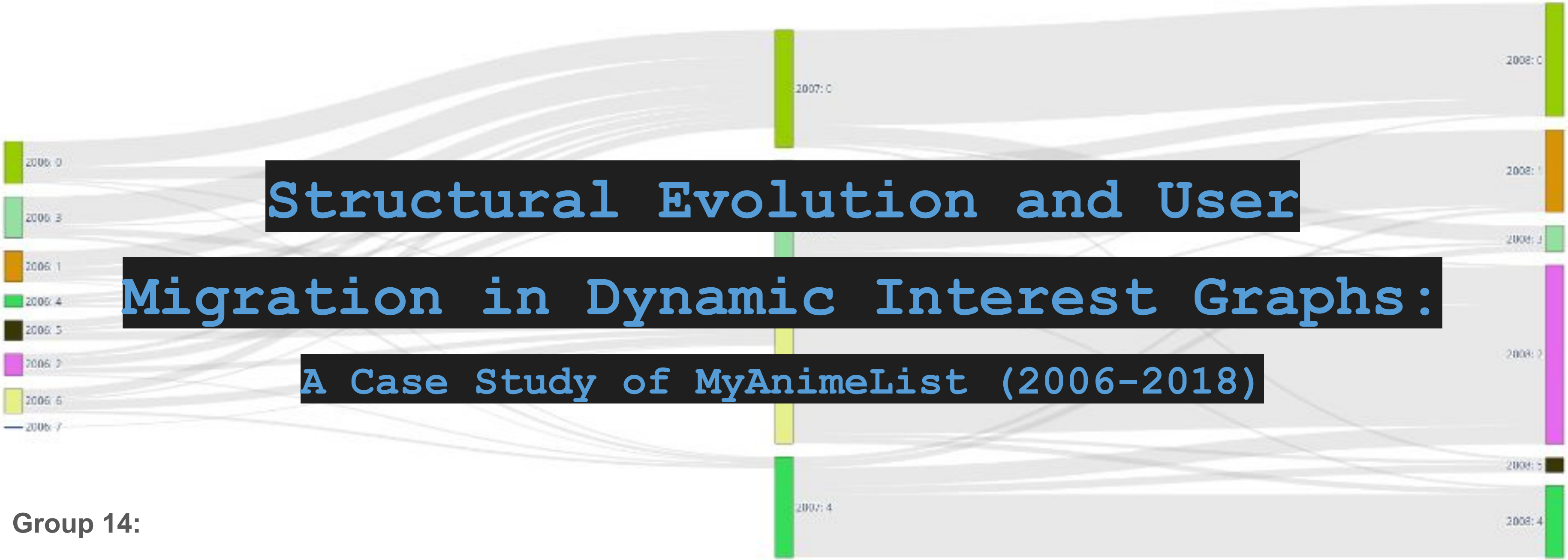


Structural Evolution and User Migration in Dynamic Interest Graphs:

A Case Study of MyAnimeList (2006-2018)

Group 14:

Iaroslav Sagan (66661),

Anna Maksymchuk (66662),

Mariia Samosudova (66663)

Report on Complex Data Analysis, 9/12/2025

Content

1. **Introduction & Framework**
   Motivation: Physics of taste & "Phase transitions" in communities. Data source & `project_cda` pipeline. Graph construction (Bipartite projection).

2. **General Network Characteristics**
   Macro-evolution of the topology (2006–2018). The "Densification vs. Diameter" paradox. Global metrics stability.

3. **Clustering & Community Dynamics**
   Community detection (Leiden algorithm). The "Fission" process: Structural separation of interests. Visualizing the landscape of subcultures.

4. **Stochastic Baselines: The Random Walker**
   Defining the Null Hypothesis (Markovian motion). Real User Trajectories vs. Random Walks. Quantifying "Intent" (Divergence from stochasticity).

5. **Predictive Modeling (ML)**
   Feature engineering: Topological vs. Semantic features. Forecasting user migration. Final Verdict: Deterministic Flow vs. Random Diffusion.

1. Motivation & Problem Formulation.

Methodology & Computational Framework.

Dynamics of Digital Affinity Networks

Motivation & Significance:

- **Shift in Social Fabric:** Modern social structures are increasingly driven by *Affinity* (shared interests) rather than *Geography*.
- **Emergence of Order:** How do individual, stochastic choices aggregate into stable, thermodynamic-like structures?
- **Stability vs. Plasticity:** What forces hold a community together (cohesion) versus what forces drive them apart (segregation)?
- **The "Digital Petri Dish":** MAL (2006–2018) allows us to observe these "phase transitions" in a closed system — from a homogeneous core to a heterogeneous, multi-cluster state.
- **Gap in Research:** Existing studies often analyze *static* snapshots. We focus on the **temporal evolution** of taste and the mechanics of user migration.

System Definition:

- Modeled as a **Dynamic Bipartite Graph**, where edges represent timestamped ratings.

Research Questions:

- **Macro-Topology:** How does the structural complexity evolve during hyper-scaling? (Does the network fracture or coalesce? densify, or expand?)
- **Mechanics of Segregation:** Why do clusters detach? What defines the "surface tension" between communities that prevents them from merging back?
- **Micro-Dynamics (user navigation):** Is the flow of users between clusters a random diffusion or a directed motion driven by structural properties? Can we predict migration between taste clusters?

Graph Construction & Computational Pipeline

Data Processing:

- **Source:** ~85k active users, ~6.5k titles (filtered for bot activity and signal sufficiency).
- **Computational Core:** Developed custom modular framework **`project_cda`** (part of MARS_1.0 repo).
- **Performance:** Leveraged **`igraph`** **C-core** for vectorized graph operations and `ijson` for streaming large-scale interaction logs.

Projection Topology (Bipartite → Monopartite):

- **Anime-Anime Network ($I-I$):**
  - Edge Weight: **Jaccard Similarity** ($J > 0.05$).
  - *Goal:* Sparse, meaningful topology; preventing the "hairball" effect.
- **User-User Network ($U-U$):**
  - Edge Weight: Co-rated items count ($w > 3$).
  - **Hub Removal:** Explicit exclusion of "Super-Hubs" (e.g., *Death Note*) to mitigate artificial clique percolation.

Result:

- A series of time-sliced networks ($G2006$, …, $G2018$) with controlled density ($\rho \approx 0.2$), optimized for community detection algorithms.

2. Topological Evolution of Projected Networks

Methodology – The Metrics of Topology

The Analytical Approach

We calculated a comprehensive suite of graph metrics to investigate the network. Moved beyond simple "counts" to understand the **Topological Shape** of the ecosystem.

Key Metrics Selected for Analysis

Focused on three core dimensions to define the network's evolution:

- **Global Connectivity (Density & Diameter) -** to determine if the network is becoming a "tight ball" (Small World) or unravelling into a "long filament" (Sprawl).

- **Navigability (Average Path Length) -** to measure the "friction" or social cost for a user to discover new content or people.
- **Local Cohesion (Clustering & Assortativity)** - to see if the network is integrated or fractured into isolated "echo chambers" and "genre bubbles."

Anime Network – The Growth Paradox

The Data: Explosive Volume

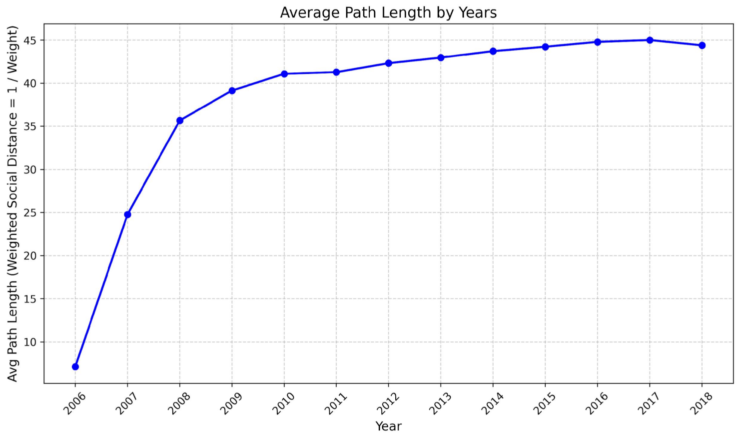
- Nodes (Anime Titles): Increased from **732** to **6,129** (↑ 737%)
- Edges (Connections): Increased from **~64k** to **~819k** (↑ 1,180%)
- Graph Density: Collapsed from **0.239** to **0.044** (↓ 82%)

The Paradox: "Hollowing Out"

- The Observation: We added over 700,000 new connections, yet the network became **less** connected.
- The Reason: The number of anime titles grew so fast that connections couldn't keep up.
- The Shift: The network transitioned from a **"Small Village"** (where everyone knows everyone) to a **"Sprawling Metropolis"** (where people stay in their own neighborhoods).

Anime Topology – Evidence of "Elongation"

The "Small World" Has Stretched into "Long Corridors"

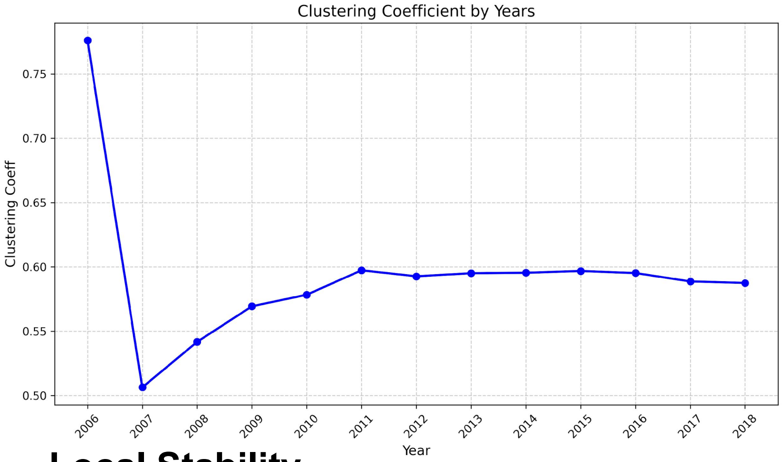


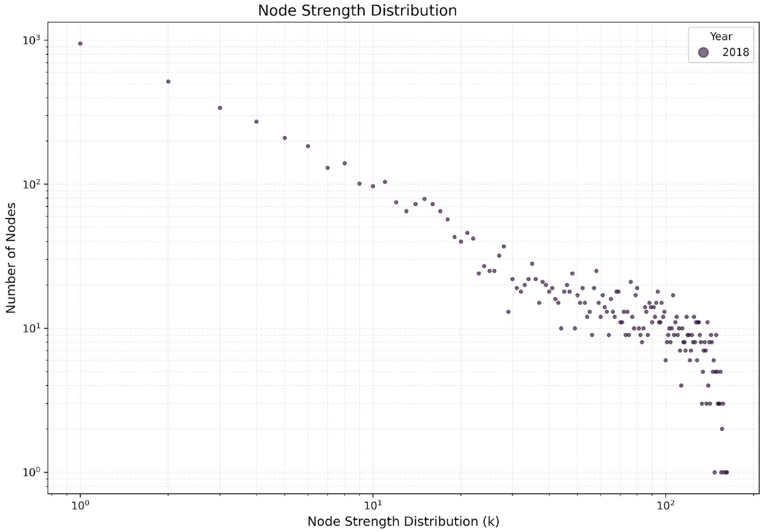
The Friction Spike
Metric: Average Path Length
The Shift: In 2006, it took **7 hops** to connect two random anime. By 2018, it took **44 hops**.
Meaning: Global shortcuts have disappeared. Navigating between genres now requires passing through dozens of intermediate titles.

The Sprawl
Metric: Network Diameter (The "Width" of the Graph)
The Shift: The distance between the two furthest points widened from **29 to 188**.
Meaning: The graph has physically stretched. Distinct taste clusters are now mathematically very far apart.

Anime Internal Structure – The “Rich Club Core”

High Local Cohesion & The Power Law





Local Stability

Metric: Clustering Coefficient

The Trend: After an initial drop, the line stabilized at a high value of **~0.59**.

Meaning: Despite the global sprawl, local neighborhoods remain tight. If you watch Anime A and Anime B, there is a high probability they are linked. The graph is built of sturdy "bubbles," not random noise.

The Backbone

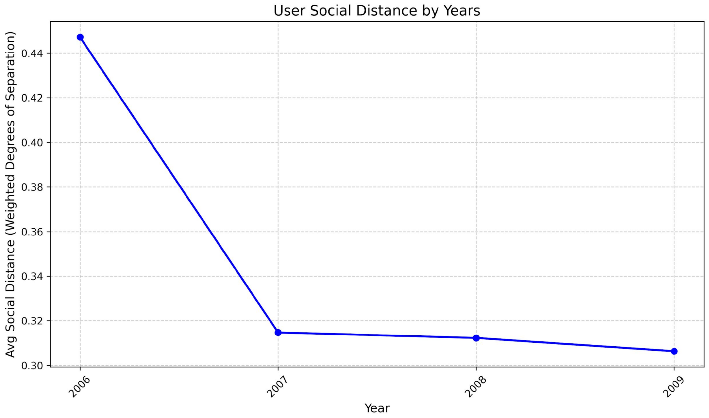
Metric: Node Strength Distribution (Log-Log Plot)

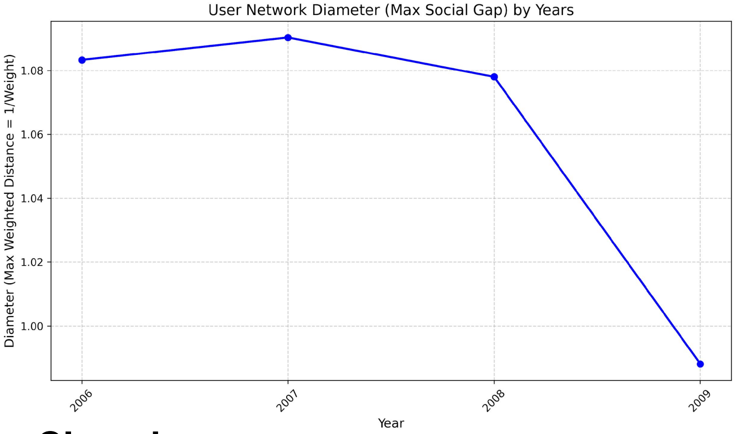
The Trend: The data follows a strict downward straight line.

Meaning: This confirms a **Power Law** distribution. The network is dominated by a number of "Mega-Hubs" (very popular shows) that have thousands of times more connections than the average title.

Insight: Rich Club Effect - a rigid backbone of

User Network – "Shrinking World" Phenomenon





Social Distance Collapsed

Metric: Average Social Distance (How "far" one user is from another).

The Trend: The line drops sharply from **0.45 to 0.31**.

Meaning: Unlike the anime graph, navigation got *easier*. As the community grew, users created shortcuts (friendships) that pulled everyone closer together.

The Gap Closed

Metric: Network Diameter (The "widest" gap between two users).

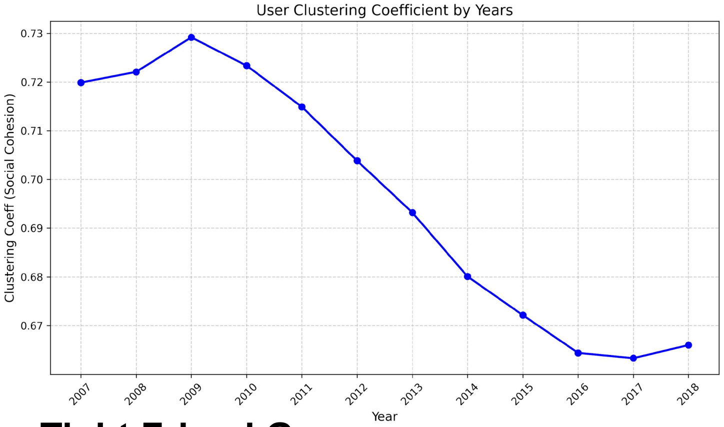
The Trend: The diameter shrank from **1.08 to less than 1.0**.

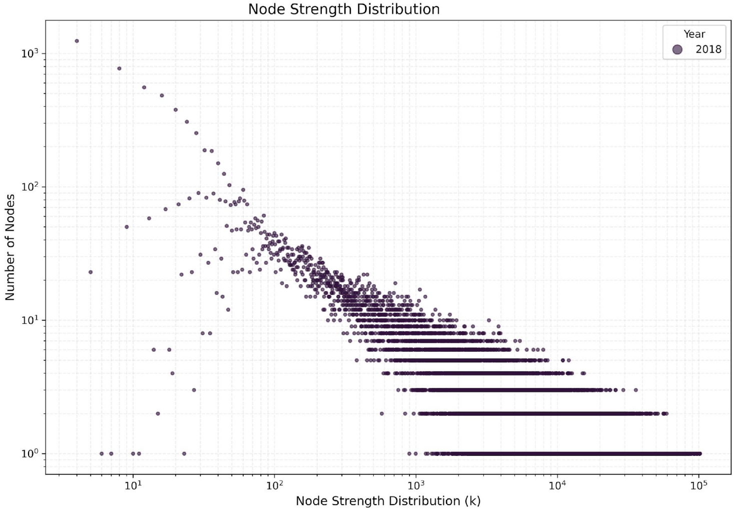
Meaning: There are no "isolated islands" of fans. Even the most distant groups (e.g., specific language speakers or niche fans) are integrating into the main core.

Insight: Global Integration - while the content library expanded and fragmented, the human community became a tighter, faster-moving "Global Village."

User Topology – The "Super-User" Glue

Highly Integrated "Global Village"





Tight Friend Groups

Metric: Clustering Coefficient (Social Cohesion)

The Trend: Peaked at **0.73** and remains very high at **0.66**.

Meaning: This is incredibly high for a social network. It proves the community isn't just a crowd of strangers; it is built on overlapping friend circles.

The "Super-Users"

Metric: Node Strength Distribution

The Trend: A straight diagonal line (Power Law).

Meaning: The network relies on a few "Super-Users" (Influencers). These few people have thousands of times more connections than the average user, acting as the "glue" that connects all the smaller friend groups together.

Insight: Structure - the community stays connected because "Super-Users" bridge the gap between

Conclusion – Two Opposite Paths

Feature	Anime Graph (Content)	User Graph (Community)
Trend	Expansion & Divergence	Densification & Convergence
Navigation	Harder (Path: 7.1 -> 44.4)	Easier (Path: 0.45 -> 0.31)
Diameter	Widening (Niches forming)	Shrinking (Subgroups integrating)
Topology	Elongated “Genre Silos"	Integrated "Small World"

Content Graph: Expansion & Fragmentation
The Shift: The library didn't just grow; it **stretched**.
The Consequence: The ecosystem elongated into specialized, distant silos. Navigating from one genre to another is now mathematically difficult (High Friction).

User Graph: Densification & Integration
The Shift: The community didn't fracture; it **tightened**.
The Consequence: New users acted as bridges. The social world "shrank," creating a hyper-connected environment where trends move instantly (Low Friction).

The Structural Paradox
Core Insight: We have built a sprawling, labyrinthine content library, but it is populated by a highly unified, 'Small World' community.

3. Mesoscale Analysis & Structural Validation

Experimental Setup & The Algorithm Battle

- **The Candidates:** Evaluated 5 algorithms (Leiden, Infomap, Eigenvector, LPA) on annual snapshots (2006–2018).
- **Evaluation Metrics: Structural:** Modularity (*Q*), **Semantic:** Genre Purity & Source Purity. **Usability:** Stability of Cluster Count (*N*).
- **The Granularity Trade-off: Label Propagation:** Underfitting (*Q≈0.03*). Too simple. **Eigenvector:** Unstable (Clusters fluctuated 4→108). **Infomap:** Hyper-segmentation. Excellent semantic purity (0.63), but produced ~150 micro-clusters.
- ***Verdict:*** We need macro-communities, not micro-fragments.

Algorithm	Modularity, Q	Genre Purity	Source Purity	Average N
Label Propagation	0.033	0.31	0.35	~ 6
Leading Eigenvector	0.301	0.50	0.45	~ 34
Infomap, T = {10, 25, 50}	max Q = 0.345	~0.62	0.54	30-150
Leiden, gamma = [0.5, 0.6, …, 5.0]	**opt Q = 0.358**	0.57	0.53	**~ 6 - 15**

The Winner: Leiden Algorithm (gamma = ~1.0)

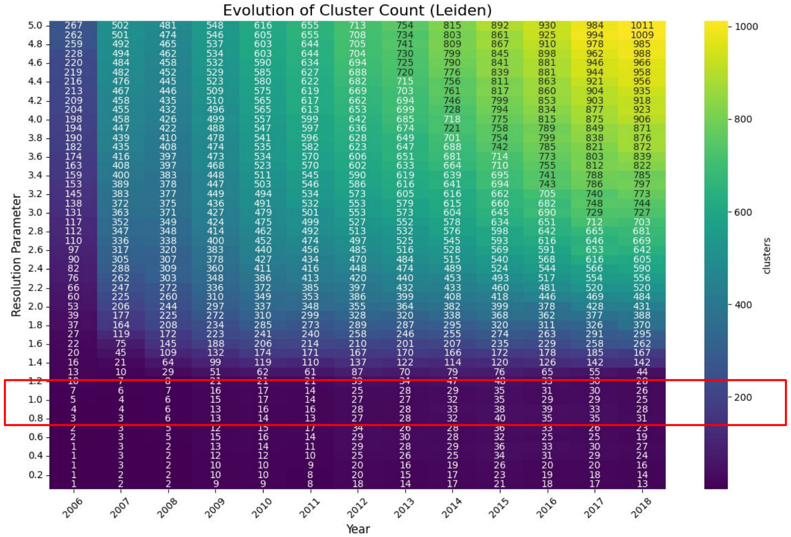
Selection: Leiden Algorithm (Modularity Optimization).
Why gamma = 1.0? The "Golden Mean":

- **Structural Integrity:** High Modularity (*$Q \approx 0.36$*), comparable to best performers.
- **Semantic Alignment:** Genre Purity (0.57) remains competitive with Infomap.
- **Interpretability:** Consistently yields ~10 stable macro-communities.

Strategic Decision:

- Prioritized **structural interpretability** over maximal semantic purity.
- Goal: Tracking migration between distinct "Continents" (Macro-genres), not "Villages" (Micro-tags).

Robustness Check: Parameter Sensitivity

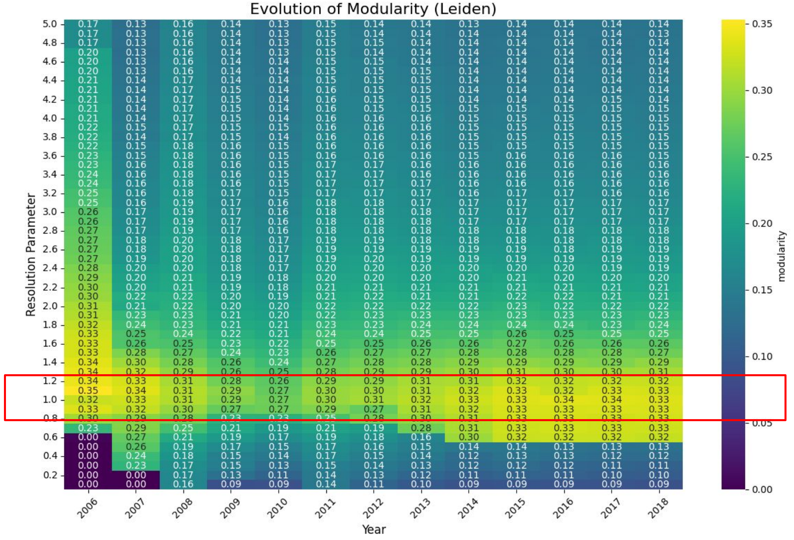


Sensitivity Analysis:

- Performed a parameter sweep for resolution γ∈[0.1,3.0] across all years.

Results (The "Plateau" Effect):

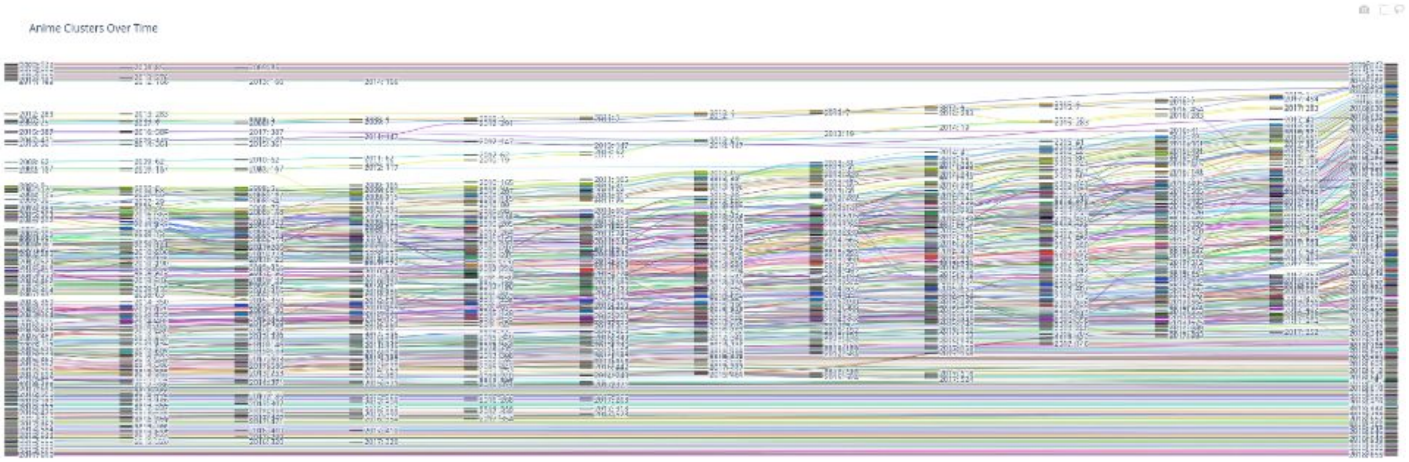
- **Modularity:** Remains robust and high in the range γ∈[0.8,1.2].
- **Cluster Count:** Stabilizes around ~10 communities in the same range.

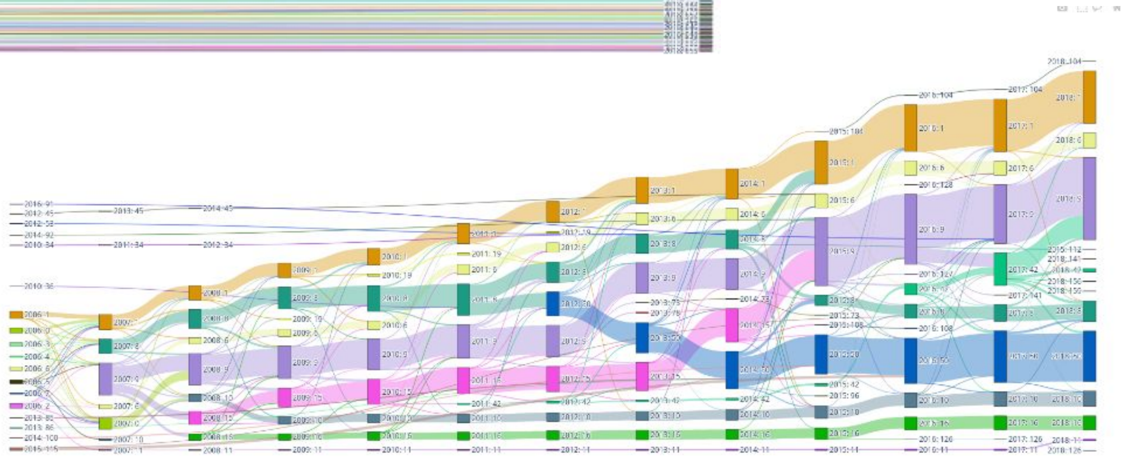


Conclusion:

- The detected structure is **topologically real**, not an artifact of a specific parameter point.
- Higher resolutions (*gamma > 1.5*) lead to artificial splintering of the network.

Leiden vs. Infomap: Sankey diagrams





4. Temporal Dynamics & User Migration Modeling. The Random Walker Approach

Random Walker experiment. Simulation Protocol

The Question: Is user migration driven purely by network topology?

- *Hypothesis:* Users follow the "path of least resistance" (strongest edge weights).

The Agent: Random Walker (Null Model)

- Operates under **Markovian Neutrality**: No memory, no taste, only structure.

Experimental Setup:

1. **Initialization:** Spawn *K=100* stochastic agents for *every* user at their 2006 position.
2. **Propagation:** Agents navigate the evolving graph snapshot by snapshot ($Gt \rightarrow G\_t+1$).
3. **Consensus:** The "Predicted Community" $c\text{^}t+1$ is determined by **Majority Vote** of the 100 agents.

Goal:

- Treat the Random Walker as a baseline classifier.
- Compare simulated trajectories (*Tsim*) vs. empirical user history (*Tuser*).

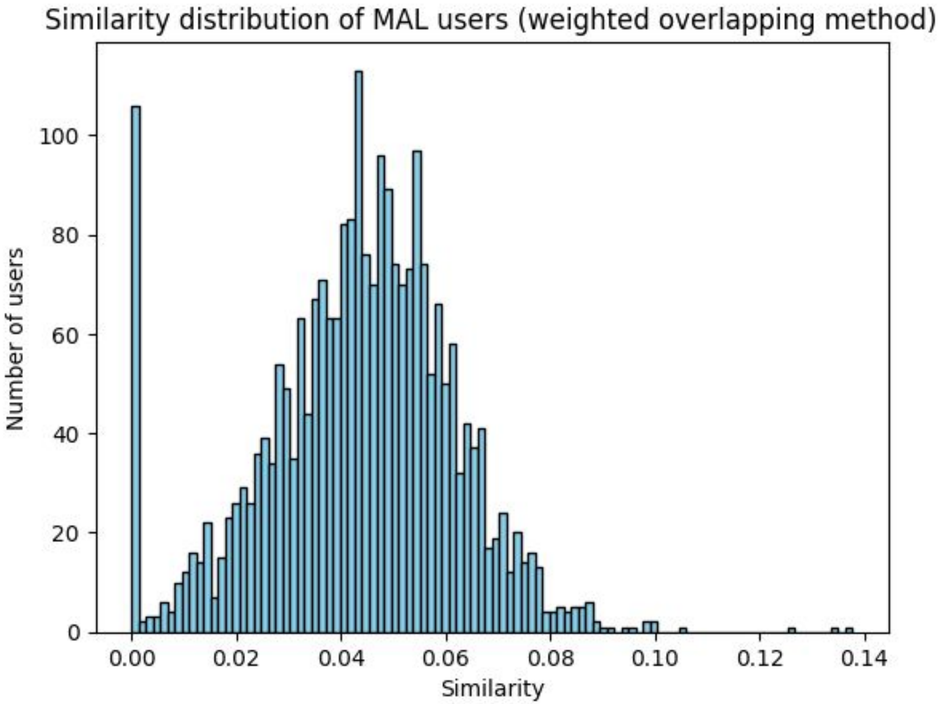
The Verdict: Structure != Destiny

Results: Total Divergence

- **Mean Jaccard Similarity:** 0.0435 (±0.0184).
- Users share only **≈ 4.3%** of their path with the topological agent.

Interpretation:

- Structure defines **Possibilities** (neighbors), not **Probabilities** (choices).
- The "Gravity" of structural hubs fails to capture human nuance.



Conclusion:

- Passive diffusion model **fails**.
- User migration is an **active choice** driven by latent factors (content, trends).
- ⇒ **Next Step:** Supervised Machine Learning.

5. User Graph. Analysis of User Migration between Communities

Predicting User Migration: Problem Formulation

1. Research Subject & Model

- **Dynamic Social Network:** The MAL user network is modeled as an **evolving graph** $G_t = (V_t, E_t)$.
- **Nodes (V_t):** Unique users.
- **Weighted Edges (E_t):** Social proximity based on shared consumption history (Affinity Network).
- **Focus:** Analyzing **transitions** between consecutive temporal snapshots (t and $t + 1$).

2. Formalizing the Task

User migration prediction (change in community membership) is formalized as a **binary classification problem**.

- *C(u, t)*: Community assignment of user *u* at time *t*.
- **Target Variable (y_u):**

$$y_u = \begin{cases} 1 \text{ if } C(u, t+1) \neq C(u, t) \\ 0 \text{ if } C(u, t+1) = C(u, t) \end{cases}$$

3. Prediction Method

- **Features:** Features extracted at time t are utilized to predict the state y_u at time $t + 1$.

Constructing User Yearly Graphs (2006–2018)

Goal: To build a series of yearly user graphs $\boldsymbol{G_{2006}}$ to $\boldsymbol{G_{2018}}$ for dynamic analysis.

To reveal genuine community structure and migration patterns, we apply three levels of **filtering**:

- **User filtering** excludes users who have watched very few anime (e.g., only 1–5 titles) or extremely many titles (e.g., binge-watchers who have watched thousands). Such users either contribute little information or create overly dense connections that obscure real community structure.

- **Anime filtering** removes overly popular titles, such as Naruto or One Piece, which are watched by a large fraction of users. These titles create super-connected hubs, making many users appear connected even if they have little else in common, which can mask meaningful migration patterns.

- **Edge filtering** removes weak connections by requiring users to share at least a minimum number of anime titles (default threshold = 3). This ensures that only significant user overlaps are included.

Community Detection and Tracking Over Time

1. The Core Task & Objective

- **Goal:** To partition social actors (users) into densely connected groups, and track changes in community membership over time to analyze user migration.
- **Objective Function:** Algorithms optimize **modularity**, a quality function that quantifies how much more densely connected nodes are *within* communities compared to connections *between* them.

2. Algorithm Selection: The Leiden Algorithm

3. Advantages of Leiden

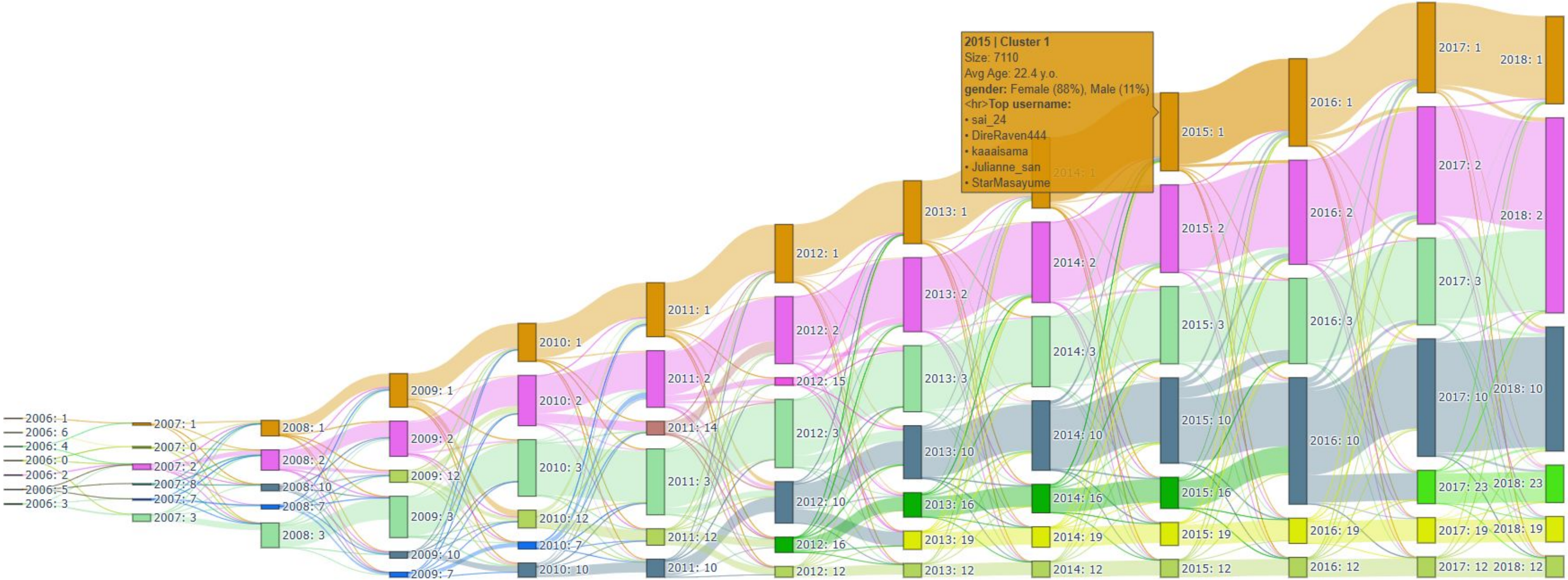
- **Scalability & Efficiency:** Near-linear scalability allows effective handling of large network sizes.
- **Quality Guarantee:** Introduces a refinement phase that **ensures all resulting communities are well-connected**, which is critical for interpreting membership as genuine social behavior.
- **Reliability:** Detected communities are more structurally meaningful and interpretable units for migration analysis.

4. Setting the Resolution Parameter

Leiden is a hierarchical algorithm, where the resolution parameter controls the size of detected communities:

- **Setting:** We set the resolution parameter = 1.
- **Rationale:** This choice provides **balanced community granularity**, avoiding overly coarse clusters (resolution < 1, macro-genres) that merge distinct groups while preventing excessively fine clusters (resolution > 1, niche noise). This setting captures meaningful structures suitable for migration analysis.

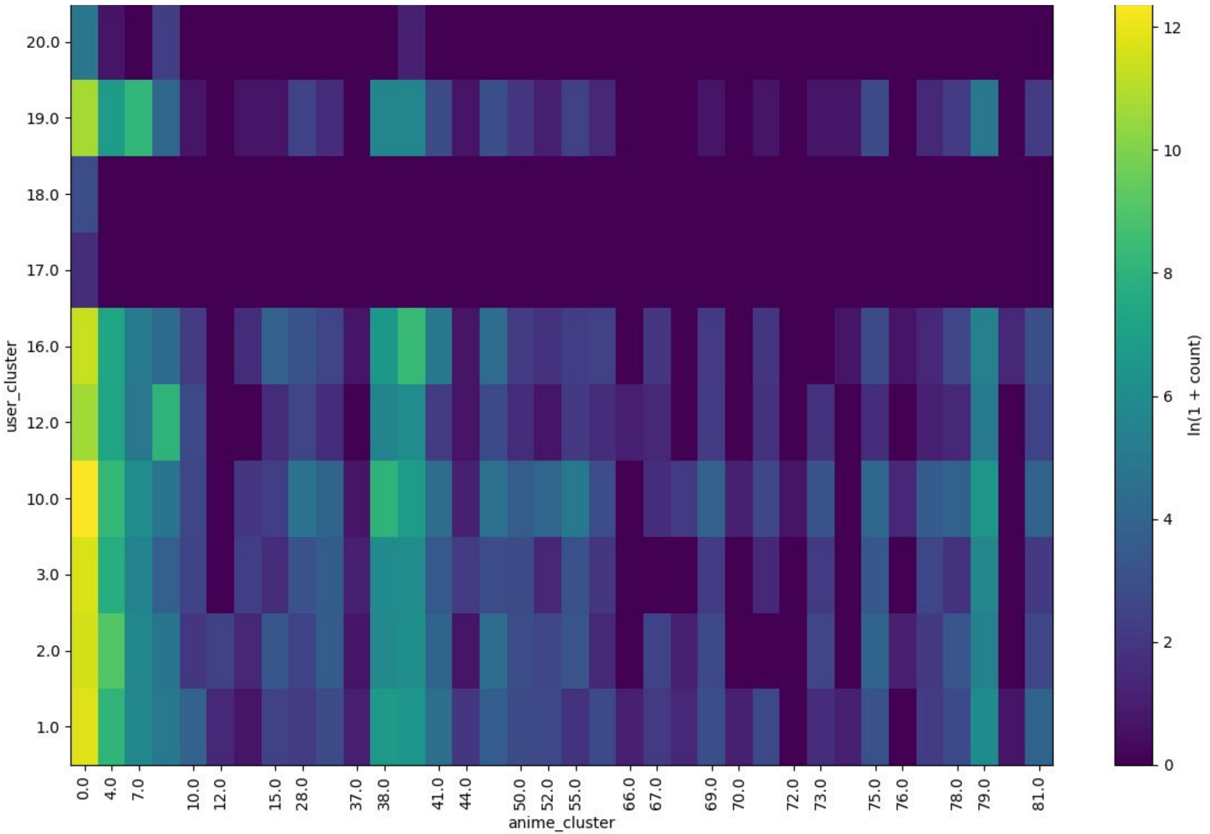
Community Detection. User Graph Evolution.



Bridging Topology and Semantics

To connect community structure (topology) with user preference (semantics), we built a **correlation matrix** between the detected user communities and specific anime genre clusters.

- The matrix is based on historical rating data, revealing which **anime categories dominate** within each user group.
- **Preference Profiling:** Uncover **community-specific preference profiles** (e.g., this user group prefers "Sci-Fi" and "Mecha").
- **Driving Factors:** Explore the **content-driven factors** underlying user migration, showing *what* content motivates users to shift communities.



Multidimensional Feature Space for Migration Prediction

To capture the factors driving migration, we constructed a comprehensive, per-user, per-year feature vector describing the user's state at time *t*. This vector integrates static, structural, and dynamic signals for predictive modeling.

Feature Categories

(1) Graph Structure (Global & Local)

- **Key Metrics:** Degree, Weighted Strength, PageRank, and K-core decomposition.
- **Rationale:** Measures user **centrality and influence** within the overall graph topology.

(2) Local Cohesion

- **Key Metric:** Weighted Clustering Coefficient.
- **Rationale:** Low values indicate **weak neighborhood integration** and a potentially **higher migration risk**.

(3) Community Embeddedness

- **Key Metric:** Intra-Community Ratio (ICR), defined as the fraction of a user's edges that connect to nodes within the same community.
- **Rationale:** Quantities the **boundary position** of a user—how strongly they are tied to their group.

(4) Temporal Dynamics

- **Key Metrics:** Delta features (delta_degree, delta_ICR, etc.).
- **Rationale:** Represents the year-over-year **change (trajectory)** in structural metrics, capturing evolving engagement.

(5) Demographics & Activity

- **Key Metrics:** Static attributes (gender, age, location) integrated with dynamic indicators (watched titles, rating fluctuations over time).
- **Rationale:** Provides non-topological context and reflects changes in **overall engagement level**.

Predictive Modeling: Why CatBoost?

We utilize **CatBoost**, a state-of-the-art gradient boosting algorithm on decision trees, due to its optimal fit for the complexities of our network data:

- **Native Handling of Heterogeneous Data:** CatBoost natively processes **categorical features** (like community membership, gender, country) alongside continuous graph metrics, avoiding complex and sparse one-hot encoding.
- **Robustness and Scale Invariance:** Highly effective with features of **widely different scales** (e.g., PageRank vs. Gender) and efficiently handles **missing values**.
- **Captures Non-Linear Interactions:** Tree-based boosting automatically captures the complex, non-linear relationships between structural position, community boundaries, and user activity dynamics.
- **Imbalanced Target Handling:** Allows for customized loss functions and class weighting to effectively manage the rare nature of **migration events**.
- **High Interpretability:** Supports native calculation of **Feature Importance** and SHAP values, crucial for understanding the structural and dynamic factors driving user migration.

Predictive Modeling: Why CatBoost?

We utilize **CatBoost**, a state-of-the-art gradient boosting algorithm on decision trees, due to its optimal fit for the complexities of our network data:

- **Native Handling of Heterogeneous Data:** CatBoost natively processes **categorical features** (like community membership, gender, country) alongside continuous graph metrics, avoiding complex and sparse one-hot encoding.
- **Robustness and Scale Invariance:** Highly effective with features of **widely different scales** (e.g., PageRank vs. Gender) and efficiently handles **missing values**.
- **Captures Non-Linear Interactions:** Tree-based boosting automatically captures the complex, non-linear relationships between structural position, community boundaries, and user activity dynamics.
- **Imbalanced Target Handling:** Allows for customized loss functions and class weighting to effectively manage the rare nature of **migration events**.
- **High Interpretability:** Supports native calculation of **Feature Importance** and SHAP values, crucial for understanding the structural and dynamic factors driving user migration.

Model Performance and Interpretation

Considering the task complexity, dataset size, and class imbalance, the CatBoost model demonstrates good performance.

Discrimination: ROC AUC = 0.915, indicating ability to separate migrating and non-migrating users.

Classification metrics:

Non-migration (0): precision 0.93, recall 0.92, F1 0.93

Migration (1, minority): precision 0.64, recall 0.68, F1 0.66

Overall accuracy: 0.88, despite class imbalance (~17% migration).

Confusion matrix: correctly detects 6,151 migrating users with controlled false positives (3,417).

The model effectively captures behavioral and network-driven patterns, making it suitable for early-warning migration detection. Thresholds can be adjusted to favor recall or precision depending on practical needs.

Model Interpretation: Feature Drivers of Migration

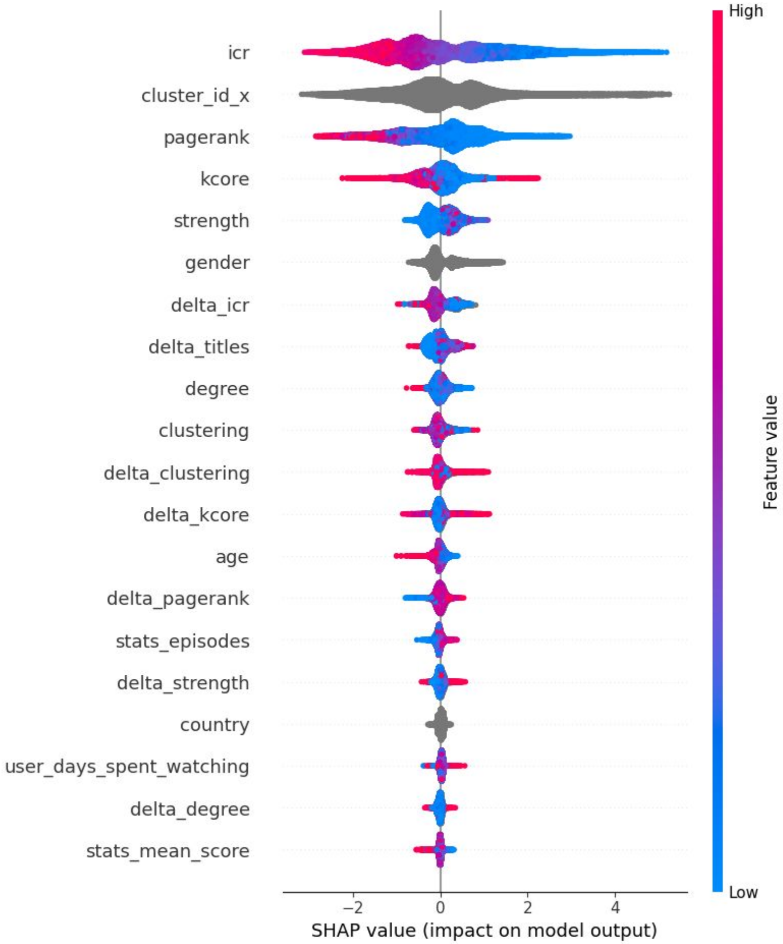
Stabilizing Force (High ICR): A high ICR value strongly contributes to a prediction of non-migration (0). This confirms that topological integration—having a high density of connections within the community—acts as a powerful anchoring effect. The user has too much invested social capital within the group to easily leave.

The Role of Global Influence (PageRank): Highly influential users (high PageRank) are less likely to migrate. This suggests that the accumulated social capital creates a significant inertia against change.

Community Instability: The current community assignment itself proved to be a strong predictor. This implies that baseline migration risks differ significantly across fandoms.

Behavioral Stability: Demographically, older users show lower migration propensities. This aligns with general sociological findings that older cohorts exhibit more stable preferences and lower rates of behavioral change than younger users.

Our investigation into user migration confirms that complex social behavior cannot be captured by singular metrics or simplistic topologica models.



Last slide…