

Recommended Analytical Workflow: Intellectual Structure and Evolution of the VIS Co-Authorship Community

This task combines three distinct analytical problems that none of the individual library workflows solves alone: (1) dynamic community detection in a co-authorship network, (2) tracking core stability, membership drift, and merge/split/dissolution events over 35 years, and (3) characterising each community's research foci from text. I therefore propose a composite workflow that adapts pieces from several library workflows — principally **Reducing Snapshots to Points**, **MobilityGraphs**, **NodeTrix**, **SOMFlow**, **EventThread**, **Human-Steered Topic Modelling (UTOPIAN)**, **StratomeX**, and **ReIEx** — into a single pipeline with a human-in-the-loop visual interface at every stage.

Phase 0 — Data Preparation

Construct one record per paper with disambiguated author IDs, year, title, abstract, keywords, and citation links. Build a full-period co-authorship graph (nodes = disambiguated authors, edges = weighted by number of joint papers) as a reference structure, analogous to the "supergraph across all time steps" used in **MobilityGraphs** to obtain a consistent spatial aggregation before temporal analysis. This full-period graph is *not* the final community structure — it only anchors later comparisons.

Phase 1 — Temporal Network Construction

Following the snapshotting logic of **Reducing Snapshots to Points**, divide 1990–2024 into overlapping time windows (e.g., 5-year windows advancing by 1–2 years) to keep adjacent snapshots smoothly related rather than abruptly different. For each window, build a weighted co-authorship subgraph containing only authors/edges active in that period. This yields a sequence of comparable network states, exactly the input structure that workflow uses for detecting stable, recurring, and outlier states.

Phase 2 — Community Detection: Anchor + Per-Window

1. **Anchor communities:** Apply a density-aware community-detection algorithm (e.g., Louvain, Leiden, or an OPTICS/DBSCAN-style approach as used for spatial aggregation in **MobilityGraphs**) to the full-period graph. This produces a stable set of "reference" macro-communities that persist across the whole dataset — the analogue of **MobilityGraphs**' spatial simplification step, which prevents weakly-connected authors from being merged just because they occasionally co-publish.

2. **Per-window communities:** Independently run the same detection algorithm on each windowed snapshot from Phase 1. This captures short-lived alliances, splinters, and newly emerging groups that the anchor step, by construction, would smooth over.

Both partitions are kept — the anchor communities give a long-term skeleton; the per-window communities give the fine-grained temporal signal needed for merge/split/dissolution detection.

Phase 3 — Cross-Time Community Matching and Evolution Events

For each pair of adjacent (overlapping) time windows, compute member-set overlap between every pair of communities (Jaccard or a weighted variant using publication counts). Represent this as a **community-flow graph**: nodes are (community, time-window) pairs; edge weight is overlap. High mutual overlap persistence; one community splitting its members across two next-period communities **split**; two communities converging into one **merge**; a community with no strong successor **dissolution**; a new community with no strong predecessor **emergence**.

This tracking logic is a simplified, set-based analogue of the tensor-decomposition "threads" in **EventThread**, where entities (here: authors), stages (here: time windows), and latent groupings are jointly modeled to reveal branching, merging, and persisting patterns. If community identities look ambiguous under simple overlap, consider EventThread's tensor approach directly: build an *author* × *community-membership* × *time-window* tensor and decompose it into latent "threads" that represent long-run community trajectories even when exact membership varies.

Visualization: Render the community-flow graph as an alluvial/stream diagram (community size over time, ribbons showing flow between periods), paired with a **Reducing Snapshots to Points**—style 2D projection in which each (community, window) node is embedded by its member/topic vector — clusters of nearby points across consecutive times indicate stable regimes, isolated points indicate anomalous/transient communities, and trails indicate transitions. This gives the analyst the "map of community states" needed to answer *how many distinct communities* exist and *when* they merge, split, or dissolve.

Phase 4 — Core Stability and Leadership Analysis

For each tracked community trajectory (a chain of matched per-window communities), compute per-author metrics across its life span:

- **Tenure:** fraction of time-windows in which the author is assigned to this community.

- **Centrality**: degree/betweenness within the community subgraph at each window.
- **Cross-membership**: number of distinct communities an author belongs to over the full period (identifies boundary-spanners vs. movers vs. loyal members).

Classify authors into **core** (high tenure + high centrality, present across many years — the "stable leadership"), **peripheral/transient** (short tenure, low centrality), and **boundary-spanning** (member of multiple communities, possibly simultaneously).

For detailed inspection of any single community/period, use **NodeTrix** (developed and validated on exactly this kind of co-authorship data): render the community as a hybrid node-link/matrix view, converting the locally dense collaboration core into a readable adjacency matrix while keeping bridging ties to other communities visible as node-link edges. This is the natural tool for testing hypotheses like "is this a cross pattern with one central PI and many loosely connected collaborators, or a dense block of mutually collaborating leaders?" — the two collaboration-pattern archetypes NodeTrix's own case study identifies in InfoVis co-authorship data.

For an author-centric view of movement between communities over time (a per-author "community timeline"), adapt the scarf-plot/event-sequence idiom from **Temporal Event Sequence Simplification** or **IVESA**: each author is a horizontal track, colored segments indicate community membership per period, gaps indicate inactivity. This directly visualizes membership overlap and drift, letting the analyst answer "how stable is the core, who moves, and when."

Phase 5 — Research Foci Extraction and Evolution

Build a term-document matrix from titles, abstracts, and keywords. Apply **Human-Steered Topic Modelling (UTOPIAN)**'s NMF-based approach rather than LDA, because its deterministic, steerable behavior is well suited to iterative refinement: the analyst can merge redundant topics, split overly broad ones, or seed new topics from representative documents/keywords, and immediately see the recomputed decomposition.

To connect topics to communities and track their evolution, follow the **Progressive Abstraction Analysis (Episodes and Topics)** pattern directly: treat each (*community, time-window*) pair as an "episode," aggregate its papers' topic-mixtures into a single topic profile, and visualize topic-profile distributions across communities and time using matrix/small-multiples displays (exactly as that workflow shows COVID mobility topics across countries and time). This answers "what are a community's major research foci, and how have they evolved" and additionally supports comparing foci across communities.

Phase 6 — Overlap and Cross-Community Comparison

To make cross-community membership overlap and its temporal change explicit (beyond the flow diagram in Phase 3), build a **pairwise overlap matrix** per time window — communities on both axes, cell values = number/fraction of shared authors — following the matrix-comparison logic of **ReIEx** (which uses a sender/receiver matrix to expose significant relational patterns in a large relational structure) and **StratomeX** (which compares multiple stratifications of the same population via overlap bands and supports ranking/querying for "stratifications similar to X" or "sets with high overlap"). This lets the analyst treat each period's set of communities as a "stratification" and directly query for high-overlap community pairs, candidates for eventual merging, or communities that quietly lost their distinctiveness.

Phase 7 — Integrated Interface and Iterative Refinement Loop

Assemble the above into a linked-view system, with the following interaction loop, modeled on the iterative "propose inspect refine recompute" logic common to **SOMFlow**, **UTOPIAN**, and **explAiner**:

1. **Overview**: alluvial community-flow diagram + 2D snapshot-embedding (Phase 3) — pick a community trajectory of interest.
2. **Structure drill-down**: NodeTrix view of the selected community at a chosen period (Phase 4) — inspect core/leader structure, bridging authors.
3. **Author timeline**: scarf-plot view of individual authors to verify stability/drift claims (Phase 4).
4. **Topic panel**: NMF topic profile for the selected community-period, editable via merge/split/seed operations (Phase 5); recompute and observe topic drift across adjacent periods.
5. **Overlap matrix / StratomeX-style query**: check whether the selected community is converging with, diverging from, or embedded inside another (Phase 6).
6. **Provenance log**: as in SOMFlow's flow graph, record every parameter choice (resolution, window size, overlap threshold, manual merges/splits of communities or topics) so the analyst's judgment calls are traceable and revisions can be compared against earlier ones.

The analyst iterates: adjusting the community-detection resolution or window size if communities look too fragmented or too coarse, correcting community-flow matches that seem implausible given known VIS-community history (e.g., a known research group), merging/splitting topics that don't align with domain knowledge, and re-running downstream steps. This mirrors the corrective, domain-knowledge-driven iteration explicit

in UTOPIAN and SOMFlow, and is essential given the explicit requirement that "the analyst will need to inspect intermediate results visually and use domain knowledge to interpret patterns and make judgements at each stage."

Summary of Library Sources Used

- **Reducing Snapshots to Points** — windowed snapshot construction; projecting states to visualize evolution/stability/transitions.
- **MobilityGraphs** — full-period "supergraph" simplification prior to temporal clustering; calendar/temporal-cluster visualization pattern.
- **NodeTrix** — hybrid node-link/matrix visualization of dense co-authorship communities; validated on this exact data domain.
- **EventThread** — tensor-based latent "thread" tracking for entities across stages; inspiration for robust community-trajectory tracking beyond simple overlap.
- **SOMFlow** — iterative guided clustering with explicit provenance/flow-graph tracking.
- **Human-Steered Topic Modelling (UTOPIAN)** — interactive NMF topic modeling with merge/split/seed operations for research-foci extraction.
- **Progressive Abstraction Analysis of Multivariate Temporal Data** — episode-to-topic abstraction and matrix/small-multiples visualization of topic evolution across entities and time.
- **StratomeX** — overlap/band visualization and query-based comparison of multiple groupings ("stratifications").
- **RelEx** — matrix-based visualization for exposing significant relational patterns in a large relational structure, adapted here for community overlap.
- **Temporal Event Sequence Simplification / IVESA** — scarf-plot/timeline idiom adapted for author-level community-membership tracks.