

Analytical Workflow: Locating the Vastopolis Outbreak Origin and Identifying its Transmission Mode

Natural-language description

The workflow proceeds in four phases, moving from raw social-media text to an actionable characterisation of *where* the outbreak began and *how* it is spreading.

Phase 1 — Extracting validated illness reports. Since no case data exist directly, illness events must first be inferred from free text. A hybrid keyword/classifier step flags microblog messages that plausibly report flu-like symptoms, producing a geolocated, timestamped "illness-report" entity set. Because extraction errors would bias every downstream spatial and temporal conclusion, the analyst inspects a sample of flagged and excluded messages in context and iteratively tunes the keyword/classifier criteria — an iterative criteria-refinement loop analogous to the feature-refinement loop in *behaviour-pattern-recognition-in-time-series* and the interactive labelling loop in *DaedalusData*.

Phase 2 — Spread overview and fine-grained origin localisation. Validated reports are spatially joined to the 13 zones and aggregated into population-normalised daily incidence, following the spatio-temporal aggregation approach of *extracting significant places from trajectories* (events-to-places). A small-multiples map sequence lets the analyst trace the spread across zones and days and form an initial impression of where and when it began. Because zone-level aggregation is too coarse to localise a plausible "site," a second step applies density-based spatio-temporal clustering directly to the raw coordinates of the earliest reports — reusing the two-stage clustering idea from the same *events-to-places* workflow — to isolate a compact, interpretable origin cluster, with the analyst iteratively tuning clustering parameters against the map context (landmarks, hospitals).

Phase 3 — Transmission-mode hypothesis testing. With an origin established, the analyst cycles through four competing transmission hypotheses, each tested against a different contextual data source: *person-to-person* (spatial contiguity/diffusion from the origin along road-network connectivity and population density), *airborne* (lagged correlation between daily wind direction/speed and the direction of new-case appearance — reusing the lagged cross-impact methodology of *Detecting Stable Cross-Impact Patterns in Bivariate Time Series*, here applied to wind vs. incidence instead of mobility vs. cases), *waterborne* (spatial clustering of cases near/along water bodies, independent of population density), and *point-source/other* (shared venue or landmark mentions in the text of early cases). For each hypothesis, coordinated map and time-series views — in the spirit of the metadata-relation-seeking multi-view design of *IVESA* and the correlation-exploration loop

of *LFPeers* — let the analyst judge whether the evidence supports, contradicts, or is inconclusive for that mode, accumulating a running evidence log across iterations until the picture is sufficiently clear.

Phase 4 — Synthesis. The localised origin, the characterised spread pattern, and the accumulated hypothesis evidence are synthesised into an explicit statement of the outbreak's probable origin and transmission mechanism(s), with implications for emergency-resource deployment (contact tracing/quarantine near a person-to-person hotspot, ventilation/air-quality response if airborne, water-supply intervention if waterborne, or facility investigation/closure if point-source).

ATWL representation

plaintext

```
workflow vastopolis-outbreak-source-and-transmission-analysis
```

```
template: define-unit
    loop(characterise visualise assess
        generate-knowledge define-unit)
    contextualise characterise contextualise
    visualise abstract
    loop(define-unit define-unit visualise assess)
    generate-knowledge
    loop(generate-knowledge characterise contextualise
        visualise abstract assess generate-knowledge)
    generate-knowledge
```

```
description: "Public-health investigation of a flu-like outbreak in
Vastopolis: extract validated illness reports from geolocated
microblogs, visualise their spatio-temporal spread across the city's
13 zones, localise a fine-grained outbreak origin via spatio-temporal
clustering, and iteratively test person-to-person, airborne,
waterborne, and point-source transmission hypotheses against weather
hydrography, road-network, and text evidence to guide resource
deployment"
```

```
# =====
# INPUT
# =====
```

artifact D_microblogs : entities

origin: given

internal structure: elementary

embedment: {set, space, time}

features:

- id: f_text

value structure: atomic

value type: text

description: "Message text content"

description: "Geolocated, timestamped microblog messages posted by residents over the outbreak period"

artifact D_zones : entities

origin: given

internal structure: region

embedment: space

features:

- id: f_population

value structure: atomic

value type: numeric

description: "Resident population of the zone"

description: "13 administrative city zones providing spatial context and population denominators"

artifact D_water_bodies : entities

origin: given

internal structure: region

embedment: space

features:

- id: f_water_type

value structure: atomic

value type: categorical

description: "Water body type (river, lake, reservoir)"

description: "Water bodies within the metropolitan area"

artifact D_road_network : entities

origin: given

internal structure: formation

embedment: space

features:

- id: f_road_type

value structure: atomic

```
    value type: categorical
    description: "Road classification (highway, arterial, local)"
description: "Highway/road network providing a connectivity structure
for person-to-person diffusion analysis"
```

```
artifact D_landmarks : entities
```

```
    origin: given
    internal structure: elementary
    embedment: space
    features:
      - id: f_category
        value structure: atomic
        value type: categorical
        description: "Landmark category (hospital, market, venue, other)"
      - id: f_name
        value structure: atomic
        value type: categorical
        description: "Landmark name"
    description: "Hospitals and other notable landmarks, used both as map
context and as candidate point-source locations"
```

```
artifact D_weather : entities
```

```
    origin: given
    internal structure: elementary
    embedment: time
    features:
      - id: f_wind_speed
        value structure: atomic
        value type: numeric
        description: "Daily wind speed"
      - id: f_wind_direction
        value structure: atomic
        value type: numeric
        description: "Daily prevailing wind direction (degrees)"
    description: "Daily city-wide weather observations over the outbreak
period"
```

```
# =====
# PHASE 1: EXTRACT AND VALIDATE ILLNESS REPORTS
# =====
```

```
artifact S_extraction_criteria : specification
```

origin: given
representation form: "keyword list and classifier threshold"
description: "Initial symptom-keyword list and confidence threshold for
flagging microblog messages as flu-like illness reports"

transform T_extract_reports :

intent: define-unit
manner: "keyword/symptom-based text classification"
input: D_microblogs, S_extraction_criteria
output: D_illness_reports
actor: hybrid
description: "Flag microblog messages plausibly reporting flu-like
symptoms and extract them as illness-report entities"

artifact D_illness_reports : entities

internal structure: elementary
embedment: {set, space, time}

features:

- id: f_keywords
value structure: list
value type: categorical
description: "Matched symptom keywords"
- id: f_confidence
value structure: atomic
value type: numeric
description: "Classifier confidence that the message reports
genuine illness"

description: "Geolocated, timestamped candidate illness reports derived
from the microblog stream"

loop L_extraction_refinement:

purpose: "Refine symptom-keyword/classifier criteria so that extracted
reports neither miss genuine cases nor include irrelevant chatter"

until: "The analyst judges the report set to be a sufficiently accurate
and complete representation of flu-like illness reports"

body:

transform T_summarise_reports :

intent: characterise
manner: "compute daily report counts and a stratified sample of
flagged and excluded messages"
input: D_illness_reports, D_microblogs

```

output: F_report_sample
actor: machine
description: "Prepare daily counts and a sample of borderline
  classifications for visual inspection"

artifact F_report_sample : feature(D_illness_reports)
  value structure: vector
  value type: {numeric, text}
  description: "Daily report counts and sampled flagged/excluded
    message texts with matched keywords highlighted"

transform T_visualise_sample :
  intent: visualise
  manner: "map and chronological list of sampled messages"
  input: F_report_sample, D_illness_reports
  output: V_report_sample
  actor: machine
  description: "Display sampled messages on the map and in a
    chronological list, with matched keywords highlighted"

artifact V_report_sample : visualisation(D_illness_reports,
  F_report_sample)
  layout: "linked map and chronological message list"
  form: "point markers on map; text list with highlighted keywords"
  encoding: "marker position from message coordinates; marker colour
    from classifier confidence; list highlighting from matched
    keyword terms"
  description: "Interface supporting inspection of extraction quality
    in geographic and textual context"

transform T_assess_extraction :
  intent: assess
  manner: "evaluate completeness and precision of the extracted
    report set against domain knowledge of symptom language"
  input: V_report_sample
  output: K_extraction_quality
  actor: human
  description: "Judge whether current criteria adequately separate
    genuine illness reports from noise"

artifact K_extraction_quality : knowledge(D_illness_reports)
  representation form: "quality judgment"

```

description: "Assessment of extraction precision/recall and whether criteria adjustment is needed"

if K_extraction_quality indicates refinement needed:

then:

transform T_revise_criteria :

intent: generate-knowledge

manner: "adjust keyword list, threshold, or exclusion terms"

input: K_extraction_quality, S_extraction_criteria

output: S_extraction_criteria_r

actor: human

description: "Revise extraction criteria based on observed errors"

artifact S_extraction_criteria_r : specification

representation form: "keyword list and classifier threshold"

description: "Updated extraction criteria"

transform T_reextract_reports :

intent: define-unit

manner: "keyword/symptom-based text classification, re-applied"

input: D_microblogs, S_extraction_criteria_r

output: D_illness_reports_r

actor: hybrid

description: "Re-extract illness reports under revised criteria"

artifact D_illness_reports_r : entities

internal structure: elementary

embedment: {set, space, time}

features:

- id: f_keywords

value structure: list

value type: categorical

description: "Matched symptom keywords"

- id: f_confidence

value structure: atomic

value type: numeric

description: "Classifier confidence"

description: "Revised illness-report set"

assign:

```

        S_extraction_criteria := S_extraction_criteria_r
        D_illness_reports := D_illness_reports_r
    else:
        exit loop L_extraction_refinement

end loop L_extraction_refinement

# =====
# PHASE 2: SPREAD OVERVIEW AND ORIGIN LOCALISATION
# =====

transform T_assign_zone :
    intent: contextualise
    manner: "map-based, point-in-polygon assignment"
    input: D_illness_reports, D_zones
    output: A_reports_zone
    actor: machine
    description: "Assign each illness report to its containing city zone"

artifact A_reports_zone : arrangement(D_illness_reports)
    context: D_zones
    principle: "point-in-polygon assignment to zone polygons"
    description: "Zone membership of every illness report"

transform T_compute_daily_incidence :
    intent: characterise
    manner: "aggregate per-zone, per-day counts and normalise by
        population"
    input: D_illness_reports, A_reports_zone, D_zones
    output: F_daily_incidence
    actor: machine
    description: "Compute a population-normalised daily incidence rate per
        zone"

artifact F_daily_incidence : feature(D_zones)
    value structure: matrix
    value type: numeric
    representation form: "time series per zone"
    description: "Daily, population-normalised illness-report incidence
        per zone over the outbreak period"

transform T_arrange_spread :

```

intent: contextualise
manner: "map-based arrangement with daily temporal sequencing"
input: D_zones, F_daily_incidence, D_illness_reports
output: A_spread
actor: machine
description: "Arrange zones geographically and sequence them by day to support a small-multiples spread overview"

artifact A_spread : arrangement(D_illness_reports)
context: D_zones
principle: "geographic zone positions crossed with daily temporal sequencing"
description: "Spatio-temporal arrangement of illness reports and zone-level incidence"

transform T_visualise_spread :
intent: visualise
manner: "small-multiples choropleth map sequence with point overlay"
input: A_spread, F_daily_incidence, D_illness_reports
output: V_spread_overview
actor: machine
description: "Display one map panel per day showing zone incidence and individual reports"

artifact V_spread_overview : visualisation(A_spread, F_daily_incidence, D_illness_reports)
layout: "chronologically ordered sequence of small map panels"
form: "choropleth zone polygons with overlaid point markers"
encoding: "zone fill intensity from normalised daily incidence; point position from report coordinates; point colour/size from symptom confidence"
description: "Overview enabling visual tracing of the outbreak's spati and temporal progression"

transform T_read_spread :
intent: abstract
manner: "perception-and-interpretation of spatial and temporal progression"
input: V_spread_overview
output: P_spread_pattern
actor: human
description: "Identify the general spread trajectory and the earliest

affected zone(s)/day(s) as candidates for the outbreak origin"

artifact P_spread_pattern : pattern(D_illness_reports, F_daily_incidence
representation form: "described spread trajectory"

description: "General direction/shape of spread (e.g. radiating
outward, corridor-following) and candidate origin zone/day"

artifact S_origin_clustering : specification

origin: given

representation form: "parameter settings"

description: "Initial spatial radius, temporal window, and minimum
point count for density-based clustering of the earliest reports"

loop L_origin_localisation:

purpose: "Isolate a compact, finer-than-zone spatio-temporal cluster
among the earliest reports that plausibly marks the outbreak's point
of origin"

until: "A stable, spatially compact, temporally coherent origin cluster
has been identified and is plausible given nearby landmarks"

body:

transform T_select_early_reports :

intent: define-unit

manner: "filter to the earliest days indicated by P_spread_pattern"

input: D_illness_reports, P_spread_pattern

output: D_early_reports

actor: human

description: "Restrict analysis to reports from the earliest phase
of the outbreak"

artifact D_early_reports : entities

internal structure: elementary

embedment: {set, space, time}

description: "Illness reports from the earliest outbreak days"

transform T_cluster_origin :

intent: define-unit

manner: "density-based spatio-temporal clustering"

input: D_early_reports, S_origin_clustering

output: D_origin_clusters, F_origin_labels

actor: machine

description: "Group early reports concentrated in space and time"

```

    into candidate origin clusters"

artifact D_origin_clusters : entities
  internal structure: group/cluster
  embedment: {space, time}
  features:
    - id: f_cluster_size
      value structure: atomic
      value type: numeric
      description: "Number of reports in the cluster"
  description: "Candidate compact origin clusters among early
    reports"

artifact F_origin_labels : feature(D_early_reports)
  value structure: atomic
  value type: categorical
  description: "Origin-cluster membership or noise designation"

transform T_visualise_origin :
  intent: visualise
  manner: "detailed local map with landmark and road overlay"
  input: D_origin_clusters, F_origin_labels, D_landmarks,
    D_road_network
  output: V_origin_map
  actor: machine
  description: "Show candidate origin clusters against landmarks,
    hospitals, and roads for plausibility checking"

artifact V_origin_map : visualisation(D_origin_clusters,
  F_origin_labels, D_landmarks)
  layout: "detailed local map"
  form: "coloured cluster points; landmark/hospital icons; road
    lines"
  encoding: "point colour from cluster membership; icon shape from
    landmark category; line style from road type"
  description: "Local map supporting assessment of cluster
    coherence and plausibility"

transform T_assess_origin :
  intent: assess
  manner: "evaluate cluster compactness, coherence, and plausibility"
  input: V_origin_map, D_origin_clusters

```

```

output: K_origin_quality
actor: human
description: "Judge whether a candidate cluster is a satisfactory
            origin, or whether clustering parameters need adjustment"

artifact K_origin_quality : knowledge(D_origin_clusters)
representation form: "quality judgment"
description: "Assessment of origin-cluster compactness, coherence,
            and plausibility"

if K_origin_quality indicates satisfactory origin cluster:
then:
    exit loop L_origin_localisation
else:
    transform T_adjust_origin_params :
        intent: generate-knowledge
        manner: "adjust spatial radius, temporal window, or minimum
                point count"
        input: K_origin_quality, S_origin_clustering
        output: S_origin_clustering_r
        actor: human
        description: "Revise clustering parameters based on
                    assessment"

    artifact S_origin_clustering_r : specification
        representation form: "parameter settings"
        description: "Updated clustering parameters"

    assign:
        S_origin_clustering := S_origin_clustering_r

end loop L_origin_localisation

transform T_finalise_origin :
    intent: generate-knowledge
    manner: "formulate-statements"
    input: D_origin_clusters, K_origin_quality, D_landmarks
    output: K_origin
    actor: human
    description: "State the confirmed outbreak origin location and time,
                noting any co-located landmark or facility"

```

```

artifact K_origin : knowledge(D_origin_clusters)
  representation form: "statements"
  description: "Confirmed origin location, onset time, and any
    co-located landmark relevant to the point-source hypothesis"

# =====
# PHASE 3: TRANSMISSION-MODE HYPOTHESIS TESTING
# =====

artifact S_candidate_hypotheses : specification
  origin: given
  representation form: "ordered list of hypothesis types"
  description: "The four candidate transmission modes to examine:
    person-to-person, airborne, waterborne, point-source/other"

artifact K_evidence_log : knowledge(D_illness_reports)
  representation form: "accumulated hypothesis assessments"
  description: "Initially empty; accumulates the evidence for/against
    each candidate transmission-mode hypothesis across iterations"

loop L_transmission_analysis:
  purpose: "Iteratively test alternative transmission-mode hypotheses
    against the observed spread pattern, weather, hydrography, road
    network, and text evidence until the evidence for the most plausible
    mode(s) is judged sufficiently strong"
  until: "All relevant hypotheses have been evaluated and the evidence
    log supports one or more transmission modes with sufficient
    confidence"
  body:

    transform T_formulate_hypothesis :
      intent: generate-knowledge
      manner: "select the next untested or inconclusive candidate
        transmission mode and specify the contextual variables and
        evidence computation needed to test it"
      input: S_candidate_hypotheses, K_evidence_log, K_origin
      output: S_hypothesis
      actor: human
      description: "Choose which transmission mode to test next and how"

    artifact S_hypothesis : specification
      representation form: "hypothesis type and evidence-computation

```

```
specification"
description: "Current transmission-mode hypothesis and the
contextual data/method needed to test it"

transform T_compute_hypothesis_evidence :
  intent: characterise
  manner: "compute hypothesis-specific evidence: spatial contiguity
and road-network proximity for person-to-person; lagged
correlation between wind direction/speed and directional case
spread for airborne; distance-to-water-body and downstream
clustering for waterborne; shared-venue/landmark text mining
among early cases for point-source"
  input: D_illness_reports, A_reports_zone, D_zones, D_weather,
D_water_bodies, D_road_network, D_landmarks, S_hypothesis
  output: F_hypothesis_evidence
  actor: machine
  description: "Compute the evidence variables relevant to the
current hypothesis"

artifact F_hypothesis_evidence : feature(D_illness_reports)
  value structure: vector
  value type: numeric
  description: "Per-report or per-zone evidence values specific to
the current hypothesis (e.g. contiguity metric, lagged wind
correlation, water-distance, venue co-mention count)"

transform T_arrange_hypothesis_view :
  intent: contextualise
  manner: "overlay hypothesis-relevant contextual layer with the
spread map and timeline"
  input: F_hypothesis_evidence, A_spread, D_water_bodies,
D_road_network, D_weather, D_landmarks
  output: A_hypothesis
  actor: machine
  description: "Combine the case-spread arrangement with the
contextual layer (wind vectors, water bodies, roads, or venues)
relevant to the current hypothesis"

artifact A_hypothesis : arrangement(D_illness_reports)
  context: S_hypothesis
  principle: "spread map and timeline overlaid with the
hypothesis-relevant contextual layer (wind field, hydrography,
```

```
road network, or venue markers)"
description: "Hypothesis-specific spatio-temporal arrangement for
visual comparison"

transform T_visualise_hypothesis :
  intent: visualise
  manner: "coordinated map overlay and correlation/time-series plot"
  input: A_hypothesis, F_hypothesis_evidence
  output: V_hypothesis
  actor: machine
  description: "Display the spread pattern against the contextual
layer and any derived correlation/time-series evidence"

artifact V_hypothesis : visualisation(A_hypothesis,
  F_hypothesis_evidence)
  layout: "map panel with contextual overlay, linked to a
correlation or time-series panel"
  form: "map overlay (vectors/polygons/lines/markers as relevant);
scatter or line chart for correlation/lag evidence"
  encoding: "overlay symbol/colour from contextual layer type; chart
axes from hypothesis evidence values"
  description: "Coordinated view supporting visual judgment of
alignment between spread pattern and candidate factor"

transform T_interpret_hypothesis :
  intent: abstract
  manner: "perception-and-interpretation of spatial alignment and
correlation strength"
  input: V_hypothesis
  output: P_hypothesis_pattern
  actor: human
  description: "Interpret whether the spread pattern visually aligns
with, or correlates with, the candidate factor"

artifact P_hypothesis_pattern : pattern(D_illness_reports,
  F_hypothesis_evidence)
  representation form: "described alignment/correlation evidence"
  description: "Observed degree and direction of alignment between
the spread pattern and the candidate transmission factor"

transform T_assess_hypothesis :
  intent: assess
```

```

manner: "judge whether evidence supports, contradicts, or is
        inconclusive for the hypothesis"
input: P_hypothesis_pattern, V_hypothesis
output: K_hypothesis_assessment
actor: human
description: "Evaluative judgment on the current hypothesis"

artifact K_hypothesis_assessment : knowledge(P_hypothesis_pattern)
representation form: "quality judgment"
description: "Support/contradiction/inconclusive verdict for the
        current transmission-mode hypothesis, with rationale"

transform T_log_evidence :
    intent: generate-knowledge
    manner: "append the current assessment to the accumulated evidence
            log"
    input: K_hypothesis_assessment, K_evidence_log
    output: K_evidence_log_r
    actor: human
    description: "Update the running record of tested hypotheses and
            their evidentiary support"

artifact K_evidence_log_r : knowledge(D_illness_reports)
representation form: "accumulated hypothesis assessments"
description: "Evidence log extended with the current hypothesis's
        assessment"

assign:
    K_evidence_log := K_evidence_log_r

if K_evidence_log indicates sufficient evidence has been gathered
    for one or more transmission modes and no untested plausible
    hypotheses remain:
    then:
        exit loop L_transmission_analysis
    else:
        transform T_select_next_hypothesis :
            intent: generate-knowledge
            manner: "choose the next candidate hypothesis to test, or
                    refine the current one, based on inconclusive or contradicte
                    evidence"
            input: K_evidence_log_r, S_candidate_hypotheses

```

```

        output: S_candidate_hypotheses_r
        actor: human
        description: "Update the remaining hypothesis agenda"

    artifact S_candidate_hypotheses_r : specification
        representation form: "revised ordered list of remaining
            hypotheses"
        description: "Updated hypothesis agenda for the next
            iteration"

    assign:
        S_candidate_hypotheses := S_candidate_hypotheses_r

end loop L_transmission_analysis

# =====
# PHASE 4: SYNTHESIS
# =====

transform T_synthesise :
    intent: generate-knowledge
    manner: "formulate-statements"
    input: K_origin, P_spread_pattern, K_evidence_log
    output: K_findings
    actor: human
    description: "Integrate the localised origin, characterised spread
        pattern, and accumulated transmission-mode evidence into an
        actionable outbreak characterisation"

artifact K_findings : knowledge(K_origin, K_evidence_log)
    representation form: "statements and recommendations"
    description: "Final characterisation of the outbreak: confirmed origin
        location and time; most plausible transmission mode(s) with
        supporting evidence and confidence; and recommended emergency
        resource deployment (e.g. contact tracing/quarantine, air-quality
        intervention, water-supply investigation, or facility closure)"

```

Sources drawn upon

- **extracting significant places from trajectories** (events-to-places): spatio-temporal aggregation of geolocated events and two-stage density clustering, adapted here for zone-level incidence and origin localisation.

- **behaviour-pattern-recognition-in-time-series** and **DaedalusData**: iterative criteria/feature-refinement loop pattern for validating text-derived event extraction.
- **Detecting Stable Cross-Impact Patterns in Bivariate Time Series** (Andrienko): lagged cross-impact methodology, adapted for testing wind vs. incidence correlation under the airborne hypothesis.
- **IVESA** and **LFPeers**: multi-view metadata-relation-seeking and correlation-exploration loop patterns underlying the transmission-mode hypothesis-testing loop.