

# Voynich Research Program — Action Plan v1

**Purpose.** Convert the current STA + harmonic framework into a clean, testable, and reproducible research plan. This document lists inputs, algorithms, experiments, success criteria, and deliverables. Tone: rigorous and falsifiable; no extraordinary claims without extraordinary evidence.

---

## 1) Scope & Working Hypotheses

- **H0 (null):** Voynich text has no globally consistent decipherment key; observed regularities arise from layout/production constraints.
  - **H1 (structure):** The script exhibits reproducible, language-like structure (Currier A/B, positional effects, token distributions) that survive holdout tests.
  - **H2 (periodicity candidate):** A layout-conditioned periodic signal near period ~17 exists in position-based sequences and is statistically significant under pre-registered controls.
  - **H3 (semantic alignment):** If any key exists, it must generalize across sections and align with illustration semantics (e.g., zodiac months only on zodiac pages).
- 

## 2) Data Assets

- **Images:** Yale/Beinecke MS 408 (IIIF). Used only for visual alignment checks.
- **Transcriptions:** EVA (Takahashi), v101; canonicalized to IVTFF-like lines.
- **Tokenization:** EVA grapheme longest-match (qo, ch, sh, aiin, etc.).
- **STA mapping:** Two-character family+member encoding; CSV-driven.
- **Metadata:** Currier A/B labels, scribal hands, section boundaries, folio ids.

**Provenance & integrity** - Keep a manifest with SHA256 for each corpus file; log any edits (IVTFF normalization, line merges/splits).

---

## 3) Tooling (CLI surface)

- `voytool ingest` → fetch standard EVA/v101 sources.
- `voytool normalize --map eva_to_sta.csv` → EVA→STA pass; emits JSONL with `{folio, line_id, raw, sta}`.
- `voytool stats` → token/family/bigram frequencies; position stats; word-length histograms.
- `voytool spectral` → build positional series (see §5) and run DFT/Lomb-Scargle; exports power spectra and bootstrap CIs.
- `voytool decode` → apply substitution tables (CSV) with longest-match grapheme tokenizer.
- `voytool evaluate` → scoring: bits-per-token (BPT), English-likeness heuristics, dictionary hits, perplexity deltas.
- `voytool concord` → KWIC around STA/EVA tokens.

**Engineering notes** - One chart per figure; default matplotlib styles; deterministic RNG seeds. - All scripts emit CSV/JSON + PNGs; never only plots.

---

## 4) Experimental Phases (pre-registered)

### Phase 0 — Setup & Sanity

- Rebuild corpora; verify counts vs. published baselines (tokens/words/lines per folio).
- Unit tests for tokenizer (grapheme boundaries) and STA mapping.

### Phase 1 — Substitution Scaffolding

- Implement longest-match grapheme tokenizer.
- Define substitution CSV schema (unidirectional; allow bigram/trigram entries).
- Score candidates on held-out folios using: BPT, word-shape conformity, function-word frequency, bigram KL divergence vs. English, and *coverage uniformity* across sections.
- **Reject** mappings that require per-section patches to “work.”

### Phase 2 — Structural Decoding

- Add position-conditioned rules (line-initial gallows, word-final variants).
- Incorporate Currier A/B priors; fit separate parameters and test cross-transfer.
- Ablations: remove each rule family and measure  $\Delta$  in BPT and error rates.

### Phase 3 — Semantics via Images

- IIIF-assisted checks: zodiac folios must contain correct month labels if any mapping claims month names.
- Herbal pages: penalize “month-like” strings; reward consistent plant-label placements only where expected.
- Align text bands to illustration regions; record false-positive rates.

### Phase 4 — Robustness & Replication

- Train/test splits by **scribal hand** and **section**.
  - Independent replication pack (scripts + tiny corpus) for third parties.
- 

## 5) Periodicity / Spectral Test (pre-registered)

**Series to analyze (per folio and concatenated):** - Tokens-per-line; grapheme family of first/last token per line; gallows presence index; line-initial/final markers; interword-lengths.

**Method:** - DFT and Lomb–Scargle (irregular spacing); report peak power at periods 2–64. - **Controls (pass/fail):** - *Line shuffle* (within folio): spike persists → pass. - *Random spacer insertion*: spike near 17 persists with reduced power → conditional pass. - *Reflow (randomized line breaks)*: spike disappears → expected fail. -

*Rotation / page order randomization*: invariant to order → pass. - *Transliteration swap (EVA vs. STA)*: quantify sensitivity. - *Architecture swap (baseline transformer vs. framework features)*: invariant should weaken under baseline.

**Statistics**: - Block bootstrap (respect within-folio autocorrelation); 95% CI for power at  $k=17$ . - Effect size: power ratio at  $k=17$  vs. median neighborhood ( $k \pm 2$ ). - Report pre-registered thresholds for “strong” vs. “weak” evidence.

---

## 6) Success Criteria

- **No-Key baseline**: reproducible structure without coherent plaintext implies  $H_0$  not falsified.
- **If claiming a key**: 1) Single rule set decodes **all** sections consistently. 2) Output exhibits grammar/vocabulary consistent with a historical language. 3) Independent teams replicate on fresh folios. 4) Peer-reviewed publication endures critique.

**Auxiliary metrics** - Cross-section BPT reduction; STA→model perplexity  $\Delta$ ; month-label accuracy on zodiac pages; false-positive penalties elsewhere.

---

## 7) Reproducibility Checklist

- Exact environment (Python version, packages); RNG seeds.
  - Manifest with SHA256 for corpora and outputs.
  - Command logs for every figure/table.
  - Small toy dataset + expected hashes for sanity.
- 

## 8) Risks & Mitigations

- **Overfitting to Currier A/B**: enforce cross-section validation.
  - **Tokenizer bias**: compare EVA vs. STA pipelines; publish failure cases.
  - **Layout artifacts**: rely on reflow control; report deltas.
  - **Ambiguous “success” metrics**: pre-register thresholds; publish negatives.
- 

## 9) Outreach Pack (ready-to-send items)

- 2-page technical appendix (PDF/Markdown).
  - Cover email templates (EN + FR) requesting expert review.
  - Minimal GitHub repo with CI that runs: `stats`, `spectral`, and a projector sanity check.
-

## 10) Immediate Deliverables (pick any subset)

- Export the appendix to PDF.
- Create a GitHub-ready skeleton (README, scripts, CI, sample data).
- Run `spectral` on a provided EVA/STA sample; ship CSV + plots.
- Package a zip with the CLI, sample data, and example outputs.

**Hand-off notes** - Keep this plan as the single source of truth; update version and manifest when any thresholds or procedures change.