THE GRUNDTVIG ESTUDIES

Making Angels Count

Documenting an AI-Assisted Computational Study of Danish Hymnals, 1740-1953

KATRINE LAIGAARD BAUNVIG

ABSTRACT*: This pamphlet documents the computational workflow behind the article "Tidens gang: Med engle og fugle på træk i det kristne forestillingsunivers i danske salmebøger 1740–1953" (Baunvig, fourthcoming), a study of formal, biblical, dogmatic, affective, pronominal, and 'ontic' change. The article combines a versioned project-level system prompt, used as a research protocol, with Python scripts for corpus transformation and quantitative analysis. Prompt publication makes the research design inspectable; numerical reproducibility additionally requires versioned code, lexicons, intermediate tables, and validation records. The pamphlet presents the corpus, cleaning pipeline, principal operationalisations, selected code, and proposed research package. It thereby exposes the decisions through which historical questions became countable objects and quantitative patterns became historical arguments.*

KEYWORDS: *Danish hymnals; computational humanities; reproducibility; large language models; prompt documentation; Python; corpus analysis; N.F.S. Grundtvig*

A Methodological Counterpart

The article *Tidens gang: Med engle og fugle på træk i det kristne forestillingsunivers i danske salmebøger 1740–1953*"(Baunvig, fourthcoming) asks how the imaginative world of authorised Danish hymnals changed across Pietism, Enlightenment, Romanticism, ecclesiastical conflict, democratisation, and secularization that characterized the period 1740-1953. Its argument combines measurements of hymn length, repertoire overlap, biblical and dogmatic language, the semantic role of angels, birds, and cyclical time), holiday affects, and first-person singular and plural forms.

A full computational account would overwhelm the article's historical argument. A separate companion belongs naturally in *The Grundtvig eStudies Pamphlets*, whose

stated purpose is to publish the raw data and methodological declarations underlying computational studies. The first two pamphlets connect datasets with the practical and hermeneutic decisions that produced them (Baunvig and Pedersen 2021; Agersnap et al. 2023).

In the following four linked objects are documented: corpus, prompt, Python scripts, and interpretive decisions. Their epistemic roles differ. The prompt structured the collaboration; scripts generated counts, tables, and figures; manual inspection resolved ambiguous forms and tested categories; historical interpretation remained the author's responsibility.

Corpus and analytical universes

The source corpus consists of four linked JSON collections with a schema and SQLite mirror (cf. Borcak forthcoming). The primary unit is the numbered hymn entry, or 'salmeforekomst', since one work may occur in several hymnals and textual versions. Prayers, readings, registers, and other material in the JSON collection were excluded from hymn-level counts.

The current corpus contains 5,946 hymn entries distributed across twelve publication objects: nine full hymnals and three supplements. After cleaning and quality control, 5,923 entries contain usable full text. The article's principal longitudinal figures compare six full hymnals in the kingdom sequence: 1740, 1778, 1798, 1856, 1898, and 1953. The supplements from 1845/46, 1873, and 1890 remain separate objects, used in corpus description and sensitivity checks rather than treated as replacement hymnals. The wider kingdom subcorpus contains 3,983 entries, of which 3,981 have usable full text. The Schleswig and South Jutland hymnals form a parallel lineage and were used as controls where lineage could affect the result.

All raw JSON files were read-only. Each principal analysis script reloaded the current source files, checked their structure, recreated the cleaning layer, and wrote derived objects to a versioned output directory. This reduced hidden state and made article claims traceable to scripts and source tables.

The System Prompt as Research Protocol

The AI-assisted research was conducted in ChatGPT Pro using GPT-5.6 Pro (GPT-5.6 Sol Pro) primarily during June and July 2026. The model was used for a) formulation and revision of operationalisations, b) drafting and debugging of Python code, c) inspection of intermediate results. All quantitative claims reported in the article were generated or independently verified by versioned Python scripts executed against the archived corpus files. The project-level system prompt was a detailed research protocol designed to prevent attractive computational patterns from hardening into historical claims before they had been tested through source criticism and alternative operationalisations. Its central rules included the following:

- "Do not rely on remembered counts, earlier temporary assumptions, or results calculated from previous versions of the data. Recompute or verify all consequential claims from the current files."
- "When complete certainty is impossible, make a defensible default explicit, proceed transparently, run a meaningful sensitivity analysis, and state the remaining uncertainty."
- "Treat actual JSON records as the empirical source of truth."
- "Never collapse hymnals, hymn entries, underlying works, and textual versions without stating why."
- "Every substantive numerical statement must be traceable to a computation, table, figure, or versioned result."
- "Separate observed result, methodological interpretation, historical hypothesis, and unresolved question."

The prompt also instructed the model to challenge the initial hypothesis. The study expected natural birds gradually to displace supernatural angels. Instead, birds gained strongly while angels also recovered after 1798; words for morning, evening, and the seasons produced the clearest long-term growth. The protocol thus helped redirect the question when the data supported a different historical development.
The complete prompt is released and attached as "Supplement S1". The excerpt above is included because prompt documentation is necessary for understanding the AI-assisted workflow. It is not sufficient for reproducing the numbers. Language-model outputs may vary across model versions and sessions; recent work on LLM pipelines similarly treats prompts, configurations, and intermediate traces as reproduction-critical artefacts (Pattnayak and Bhatia 2026). Numerical reproducibility therefore depends on the executable scripts, environment and file-hash manifest, versioned lexicons, and saved intermediate and figure-source tables supplied in Supplements S2–S5. The decisions by which exploratory results were accepted, revised, superseded, or discarded are documented separately in Supplement S6.

Platform	OpenAI ChatGPT Pro, desktop
Model	GPT-5.6 Pro / GPT-5.6 Sol Pro
Access period	June–July 2026
Project protocol	Computational Historical Discovery Protocol
Corpus	Versioned JSON files with recorded hashes
Computation	Versioned Python scripts and package environment
Model function	Operationalisation, code assistance, testing, interpretation, and drafting
Evidential status	Python outputs, not model-generated numbers, constitute the quantitative evidence

Table 1. *Project Metadata – documented in the Supplement material*

From Raw JSON to Analysis-Ready Text

The cleaning pipeline was conservative. Unicode was normalised with NFKC; tags, page markers, running headers, melody instructions, uncertainty labels, and identifiable machine commentary were removed from the analysis copy but retained in the raw source. A few damaged records required documented start or end anchors. Entries with fewer than twenty alphabetic tokens, or placeholder text, were excluded. No raw record was deleted.

A shortened extract from the self-contained pronoun-analysis script shows the pattern:

```
def read_json_entries(path):
    with path.open(encoding="utf-8") as f:
        obj = json.load(f)
    if "entries" not in obj:
        raise ValueError(f"Expected entries in {path}")
    return obj["entries"]

hymns = read_json_entries(ROOT / "hymns.json")
assert len(hymns) == 5946

clean_rows = []
for hymn in hymns:
    cleaned, actions = clean_text(hymn)
    n_tokens = len(alpha_tokens(cleaned))
    clean_rows.append({
        "hymn_id": hymn["id"],
        "clean_full_text": cleaned,
        "usable_text": n_tokens >= 20
    })

usable = pd.DataFrame(clean_rows)
usable = usable[usable["usable_text"]]
assert len(usable) == 5923
```

Assertions caused a script to fail when a count, merge, denominator, or aggregate no longer matched the expected corpus state. Manifests recorded file hashes, package versions, random seeds, cleaning and dictionary versions, and outputs.

Tokenisation preserved historical spelling for quotation but used case-folded and cautiously normalised shadow text for counting. The analyses never concatenated titles or first lines with full text, since this would duplicate material already present at the beginning of many hymns. Stanza and line analyses used the cleaned layout; the 1953 transcription required a separate, validated repair rule because page and column breaks occasionally appeared as false stanza boundaries.

Turning Historical Questions into Measurements

The analyses were transparent indicators rather than universal classifiers. Repertoire continuity was measured through exactly matching normalised first lines. This gives a reproducible lower bound: it misses hymns whose opening was revised, but it does not silently merge doubtful works. Work-level checks using origin_id were used as sensitivity analyses rather than treated as infallible identity resolution.

Biblical and dogmatic content was divided into three layers: A curated dictionary identified named biblical persons, places, objects, groups, and events; broad terms

such as *God*, *Jesus*, and *heaven* were excluded because they do not evoke a specific scene. Eleven tightly specified families registered explicit dogmatic language. Textual proximity to the Bible was measured through sequences of five or more consecutive words also found in Danish Bible translations from 1819, 1871, 1907, or 1931. The measure therefore identifies candidates for quotation and close paraphrase, not free allusion. A secondary split assigned unambiguous persons and places to the Old or New Testament while leaving shared or uncertain referents unassigned.

Angels were registered through controlled forms and compounds beginning with engel- or engle-. Birds were identified through fugl- and an expanded species dictionary, followed by contextual checks of ambiguous forms such as *duer* and *svale*. Cyclical time comprised words for morning, evening, midnight, dawn, spring, summer, harvest, autumn, and winter; the broader words *day* and *night* were excluded to keep the measure semantically narrow.

Affect analysis used a manually audited lexicon of historical forms grouped into joy, love, hope/trust/consolation, longing, grief, fear, shame, and anger. Counts were calculated per 10,000 words and as relative profiles within historically delimited Christmas, Easter, and Pentecost sections. The results describe explicit affect vocabulary, not the total emotional effect of a hymn.

Pronoun analysis counted the paradigms jeg/mig/min/mit/mine and vi/os/vor/vort/vore/vores. In the final classification, a hymn was singular-dominant or plural-dominant according to which family occurred more often; exact ties and hymns without first-person forms remained separate. The underlying function was simple and inspectable:

```
def count_family(text, singular, plural):
    counts = Counter(tokenize(text))
    sg = sum(counts[x] for x in singular)
    pl = sum(counts[x] for x in plural)
    return {
        "sg": sg,
        "pl": pl,
        "dominance": "sg" if sg > pl else
                     "pl" if pl > sg else "tie"
    }
```

Simple code does not imply simple interpretation. A grammatical *I* can be sung by an entire congregation, and a *we* can be read privately. The measure concerns textual subject positions, not observed reception.

Validation, Sensitivity, and Historical Judgement

For each major finding, I repeated the analysis after changing at least one consequential but defensible analytical choice: the dictionary, threshold, denominator, corpus boundary, or unit of analysis. The primary specification was selected on historical and linguistic grounds; the alternatives were chosen because another competent researcher might reasonably have defined the phenomenon differently. This was therefore not a search for the specification that produced the strongest result; it was a test of whether the historical interpretation depended on one particular coding decision. A finding was treated as robust when its direction and substantive meaning remained

stable, even if the exact values changed. Where an alternative specification altered the conclusion materially, the claim was narrowed and the dependency reported.

This procedure is especially important for historical text analysis because the phenomena under investigation are not directly observable variables. "Bird," "dogmatic core," "biblical language," and "collective voice" become countable only after decisions have been made about spelling variants, ambiguous forms, lexical boundaries, textual units, and thresholds. A single operationalisation can easily conceal such decisions beneath an apparently objective number. Sensitivity analysis makes their consequences visible. It helps distinguish a historical pattern from an artefact of dictionary design, corpus composition, text length, work duplication, or one unusually influential group of hymns. It does not remove interpretation; it disciplines interpretation by showing which conclusions survive more than one plausible translation of a historical question into code.

In practice, lexical matches were exported with hymn identifiers and textual context for manual inspection, and broad dictionaries were compared with narrower, higher-precision versions. Motifs were measured both as the proportion of hymn entries containing at least one match and as mentions per 10,000 words. Text-near biblical language was tested at different sequence-length thresholds; repeated underlying works were collapsed in work-level sensitivity analyses; and pronoun patterns were recalculated under both exclusive and dominance-based classifications. The principal kingdom hymnal sequence was compared with the Schleswig–Southern Jutland lineage where this could test whether an apparent chronological development was instead branch-specific. Fixed random seeds were used for stratified audits, bootstrap intervals, and permutation tests. Figures were generated from saved intermediate tables rather than from values entered manually into plotting code.

The procedure followed the project protocol's requirement that an unexpected pattern remain provisional until it had been reproduced, checked against the underlying records, tested under an alternative operationalisation, confronted with disconfirming cases, and assessed for historical plausibility. Executable scripts, computational environments, lexicons, and figure-source tables are supplied in Supplements S2–S5; revisions, rejected specifications, and superseded results are recorded in Supplement S6. The objective is thus neither mechanical objectivity nor unlimited methodological variation, but inspectable historical judgement: a record of how interpretive choices entered the analysis and how strongly the principal findings depend upon them.

These procedures locate interpretation rather than remove it. Following Joyeux-Prunel's argument for post-computational reproducibility, computational patterns were tested against hymn contexts, book history, editorial lineages, and close readings rather than allowed to stand alone (2024).

List of Supplementary Material

For documentation I have included the following as appendixes to the pamphlet:

- **S1:** the complete versioned project prompt;

- **S2:** the Python scripts used for article-ready results;
- **S3:** an environment and file-hash manifest;
- **S4:** angel, bird, temporal, biblical, dogmatic, affect, and pronoun lexicons;
- **S5:** figure-source tables, concordance samples, and validation files;
- **S6:** a decision log marking exploratory, superseded, and article-ready results.

References

Agersnap, Anne, Lea Wierød Borčak, Thomas Husted Kirkegaard, and Katrine Baunvig. 2023. *Danish Communal Singing Culture in the 2020ies: A Survey of Singing Habits among Adult Danes in Denmark*. Grundtvig eStudies Pamphlet 2. Aarhus: Center for Grundtvig Studies.

Baunvig, Katrine Laigaard. Forthcoming. "Tidens gang. Med engle og fugle på træk i danske salmebøgers forestillingsunivers." In: *Den Danske Salmebog. Nye Perspektiver*, Kristoffer Garne, Fedja Wierød Borčak, Katrine Laigaard Baunvig & Lea Wierød Borčak (eds.). Copenhagen: Nordic Academic.

Baunvig, Katrine, and Stine Kylsø Pedersen. 2021. *The Dana Repository: Plotting the 1,138 "Danas" Located in the Danish Royal Library's Digital Newspaper Archive 1800–1870*. Grundtvig eStudies Pamphlet 1. Aarhus: Center for Grundtvig Studies.

Borčak, Fedja Wierød. Forthcoming. "Den digitale salmebog: fra kodeks til database". In: *Den Danske Salmebog. Nye Perspektiver*, Kristoffer Garne, Fedja Wierød Borčak, Katrine Laigaard Baunvig & Lea Wierød Borčak (eds.). Copenhagen: Nordic Academic.

Joyeux-Prunel, Béatrice. 2024. "Digital Humanities in the Era of Digital Reproducibility: Towards a Fairest and Post-Computational Framework." *International Journal of Digital Humanities* 6: 23–43. https://doi.org/10.1007/s42803-023-00079-6.

Pattnayak, Priyaranjan, and Apoorv Bhatia. 2026. "ReproEvalCard: A Reporting Standard for Reproducible Evaluation of LLM Pipelines." In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics, Volume 2: Short Papers*, 238–249. Association for Computational Linguistics. https://doi.org/10.18653/v1/2026.acl-short.22.

Sandve, Geir Kjetil, Anton Nekrutenko, James Taylor, and Eivind Hovig. 2013. "Ten Simple Rules for Reproducible Computational Research." *PLoS Computational Biology* 9 (10): e1003285. https://doi.org/10.1371/journal.pcbi.1003285.