Toward Agentic Content Analysis: A Reflexive Account of Human-AI Collaborative Research
- Abstract
- 1. Introduction
- 2. CommDAAF: A Framework for LLM-Assisted Content Analysis
- 3. The Research Workflow: End-to-End Documentation
- 4. Human-Agent Dialogue: The Conversational Research Process
- 5. What Went Wrong: A Taxonomy of Failures
- 6. Ten Practices for Agentic Content Analysis
- 7. Discussion: Toward a Research Program
- 8. Conclusion
- References
- Appendix A: Full CommDAAF Coding Prompt (v0.5)
- Appendix B: Conversation Log Excerpts
- Appendix C: Failure Documentation Template
- Appendix D: CommDAAF Quick-Start Checklist

Toward Agentic Content Analysis: A Reflexive Account of Human-AI Collaborative Research

Authors: [To be determined]

Target Journal: *Computational Communication Research* / *Communication Methods and Measures* / *Journal of Communication*

Status: Preprint Draft v1.0
Date: February 2026

Abstract

Large language models (LLMs) are rapidly transforming content analysis in communication research, yet methodological guidance remains fragmented. This paper presents a reflexive account of developing and implementing an agentic content analysis workflow—one in which AI agents operate with substantial autonomy while researchers provide strategic direction. Drawing on our experience analyzing 380 #MahsaAmini tweets across three LLMs, we document the full workflow from study design through analysis, including both successes (substantial inter-model agreement, κ = 0.633) and failures (data contamination, inappropriate statistical models, oversimplified prompts). We introduce CommDAAF (Communication Data Analyst Augmentation Framework), a methodological scaffold for LLM-assisted content analysis, and extract ten concrete practices for researchers seeking to implement agentic approaches. Our central argument is that agentic content analysis requires not less human judgment but different human judgment—shifting from coding reliability to prompt engineering, model selection, and quality control. We offer this account as an experimental step toward formalizing human-AI collaborative research methods.

Keywords: content analysis, large language models, agentic AI, computational methods, research methods, CommDAAF

Word count: ~8,000

---

1. Introduction

Content analysis has been the workhorse method of communication research for over half a century (Krippendorff, 2018). Yet the method faces persistent challenges: it is labor-intensive, difficult to scale, and dependent on trained human coders whose time and attention are finite. The emergence of large language models (LLMs) capable of sophisticated text understanding has prompted excitement—and anxiety—about automating content analysis (Gilardi et al., 2023; Törnberg, 2023).

Early work demonstrates that LLMs can achieve human-level reliability on many coding tasks (Ziems et al., 2024), sometimes exceeding human agreement while dramatically reducing cost and time (Heseltine & Clemm von Hohenberg, 2024). Yet methodological guidance remains nascent. How should researchers design prompts? How many models should be used? How do we validate LLM coding without circular human oversight? What happens when models disagree?

This paper addresses these questions through a different approach: reflexive documentation of a complete research project employing what we term “agentic content analysis”—a workflow in which AI agents operate with substantial autonomy across the research pipeline while human researchers provide strategic direction, quality control, and theoretical interpretation. Rather than presenting only successful outcomes, we document our full journey, including the false starts, corrections, and iterative refinements that characterize real research practice.

Our empirical case is a study of framing and virality in the #MahsaAmini movement, analyzing 380 tweets across seven theoretically derived frames using three LLMs (Claude, GLM-4.7, Kimi K2.5). But our contribution is primarily methodological: demonstrating how human-AI collaboration can be structured, what can go wrong, and how to build in safeguards.

We introduce CommDAAF (Communication Data Analyst Augmentation Framework), a methodological scaffold adapted for communication research contexts, and extract concrete practices for researchers implementing similar approaches. Our central argument is that agentic content analysis does not eliminate human judgment but transforms it—from line-by-line coding decisions to higher-order choices about prompts, models, and quality control.

1.1 What is “Agentic” Content Analysis?

The term “agentic AI” refers to systems that can pursue goals with substantial autonomy, making decisions and taking actions without continuous human oversight (Shavit et al., 2023). In research contexts, agentic systems differ from simple automation in three ways:

1. **Autonomy**: Agents can complete multi-step tasks (e.g., “code these 400 posts for framing”) without instruction at each step
2. **Judgment**: Agents make contextual decisions when encountering ambiguous cases
3. **Iteration**: Agents can review their own work, identify problems, and propose corrections

Applied to content analysis, agentic approaches shift the researcher’s role from coder to orchestrator. Rather than coding individual texts, researchers: - Design coding schemes and translate them into prompts - Select and configure models - Define quality control procedures - Interpret outputs and identify systematic errors - Make theoretical sense of findings

This is not “pushing a button”—it is a different kind of intellectual labor with distinct challenges and failure modes.

1.2 The Promise and Peril of LLM Content Analysis

LLM-based content analysis offers compelling advantages:

Scale: What takes human coders weeks can be completed in hours **Consistency**: Models apply the same “judgment” to each text without fatigue **Cost**: API costs are typically orders of magnitude lower than human coder labor **Reproducibility**: Given the same prompt and model, outputs are deterministic (at temperature=0) **Multilingual**: Modern LLMs handle multiple languages without separate codebooks

Yet risks are equally significant:

Opacity: We cannot fully inspect how models reach coding decisions **Bias**: Models inherit biases from training data that may systematically distort coding **Hallucination**: Models may confidently produce incorrect outputs **Prompt sensitivity**: Small changes in wording can substantially alter results **Validity**: High reliability does not guarantee valid measurement of theoretical constructs

These risks do not preclude use—human coders also exhibit biases, fatigue, and inconsistency—but they demand different quality control strategies. The goal is not to eliminate human judgment but to deploy it more strategically.

---

2. CommDAAF: A Framework for LLM-Assisted Content Analysis

CommDAAF (Communication Data Analyst Augmentation Framework) provides methodological scaffolding for LLM-assisted research. Adapted from the Data Analyst Augmentation Framework (DAAF Contribution Community, 2025), CommDAAF specifies procedures for prompt design, multi-model validation, reliability assessment, and quality control tailored to communication research contexts.

2.1 Core Principles

CommDAAF is organized around five principles:

1. Theoretical grounding: Coding schemes must derive from substantive theory, not emerge post-hoc from model outputs. Frames, constructs, and categories should be defined before prompting.

2. Explicit decision rules: Ambiguous cases must be anticipated and resolved through explicit hierarchical rules, not left to model “judgment.”

3. Multi-model validation: Single-model coding is insufficient. Multiple models provide internal reliability checks and reveal construct-specific vulnerabilities.

4. Transparent reliability: Frame-specific (not just aggregate) reliability must be reported. Low-reliability constructs require flagging or exclusion.

5. Tiered claims: Findings should be calibrated to validation level—exploratory (LLM-only), provisional (multi-model), or confirmatory (human-validated). Critically, inter-model agreement establishes reliability (consistency) but not validity (accuracy). High agreement among models that systematically misinterpret a construct is worse than human disagreement that surfaces genuine interpretive differences. Validity requires comparison to human expert judgment.

2.2 The CommDAAF Prompt Structure

A CommDAAF-compliant coding prompt includes:

```
1. TASK DEFINITION
   - What the model is being asked to do
   - Output format specification (JSON schema)

2. CONSTRUCT DEFINITIONS
   - Each coding category with explicit definition
   - Theoretical grounding for categories

3. DECISION RULES
   - Hierarchical priority when categories overlap
   - Explicit resolution of anticipated ambiguities

4. ANCHORS AND EXAMPLES
   - Prototypical examples for each category
   - Counter-examples (what does NOT qualify)
   - For multilingual content: examples in each language

5. EDGE CASE HANDLING
   - Mixed-language content
   - Ambiguous or borderline cases
   - Missing or uncodable content
```

This structure ensures consistency across models and makes coding decisions transparent and auditable.

2.3 Validation Tiers

CommDAAF distinguishes three validation tiers with corresponding claim strengths:

Tier	Validation	Claim Strength	Use Case
□ Exploratory	Multi-model agreement only	Hypothesis-generating	Pilot studies, pattern discovery
□ Provisional	Multi-model + human spot-check	Tentative conclusions	Working papers, conference presentations
□ Confirmatory	Multi-model + systematic human validation	Publication-ready	Peer-reviewed articles

Our #MahsaAmini study operates at the exploratory tier (□), meaning findings should be treated as hypothesis-generating rather than confirmatory. This is appropriate for pilot work but would require human validation for definitive claims.

---

3. The Research Workflow: End-to-End Documentation

We now document the complete workflow of our #MahsaAmini study, including both planned and emergent steps. This account is deliberately transparent about missteps, as these reveal where quality control is most needed.

3.1 Phase 1: Study Design (Human-Led)

Day 1, Morning

The project began with a theoretical question: What makes protest content go viral? Drawing on framing theory (Entman, 1993) and moral contagion research (Brady et al., 2017), we hypothesized that emotionally charged frames would predict higher engagement.

We developed a seven-frame typology grounded in social movement literature: - SOLIDARITY (collective identity) - INJUSTICE (perpetrator-focused blame) - CONFLICT (active clash) - HUMANITARIAN (victim-focused suffering) - HOPE (optimistic future) - INFORMATIONAL (neutral updates) - CALL_TO_ACTION (mobilization appeals)

What the agent did: At this stage, the AI agent provided literature summaries, suggested frame definitions, and helped operationalize theoretical constructs. The agent drafted initial frame definitions based on citations to Gamson (1992), Snow and Benford (1988), and Semetko and Valkenburg (2000).

What the human did: The researcher selected which frames to include, resolved definitional ambiguities, and made theoretical judgments about construct validity.

3.2 Phase 2: Data Preparation (Agent-Led)

Day 1, Afternoon

The dataset (originally 719 tweets) required preprocessing. The agent: - Loaded raw data from CSV files - Calculated composite engagement scores: `log(RT+1) + log(likes+1) + log(quotes+1)` - Implemented stratified sampling by engagement tier - Split data into coding batches

First failure: Data contamination

The original dataset combined two distinct events: #MahsaAmini (Iran protests) and Ukraine-related content. The agent did not initially detect this mixing because both datasets used similar hashtag collection methods.

Discovery: During coding review, we noticed tweets about “Kyiv” and “Zelensky” appearing in ostensibly #MahsaAmini batches.

Resolution: We separated the datasets, creating distinct #MahsaAmini (n=380) and Ukraine (n=339) samples. Twenty contaminating posts were removed from the #MahsaAmini sample.

Lesson learned: Agents can execute data operations flawlessly while missing semantic inconsistencies. Human review of sample content—not just metadata—is essential.

3.3 Phase 3: Prompt Development (Iterative Human-Agent)

Day 1, Evening – Day 2

The initial coding prompt was minimal:

```
Classify each tweet into one of these frames:
SOLIDARITY, INJUSTICE, CONFLICT, HUMANITARIAN,
HOPE, INFORMATIONAL, CALL_TO_ACTION

Also code valence (positive/negative/neutral)
and arousal (low/medium/high).
```

Second failure: Oversimplified prompts

The first model (Kimi K2.5) coded 400 posts using this simplified prompt. Results showed reasonable face validity, but when we compared against Claude's coding (which used expanded definitions), agreement was only moderate.

Discovery: Reviewing disagreements revealed systematic differences —Kimi under-coded CONFLICT and over-coded HUMANITARIAN compared to Claude.

Resolution: We developed a full CommDAAF-compliant prompt including: - Explicit definitions for each frame - Hierarchical decision rules (e.g., “INJUSTICE > HUMANITARIAN when perpetrator is named”) - Valence anchors with examples in Persian, English, and Arabic - Arousal calibration from “calm reporting” (low) to “urgent alarm” (high) - Mixed-language handling instructions

The full prompt expanded from ~100 words to ~2,500 words.

Lesson learned: Prompt length correlates with coding consistency. The added token cost (~$0.02 per post with expanded prompt) is trivial compared to the reliability gains.

3.4 Phase 4: Multi-Model Coding (Agent-Led)

Day 2, Afternoon – Day 3

We deployed three models for independent coding:

1. **Claude Opus 4.5** (Anthropic) – Flagship Western model
2. **GLM-4.7** (Zhipu AI) – Chinese model with strong multilingual capability
3. **Kimi K2.5** (Moonshot AI) – Chinese model optimized for long context

Each model coded all 380 posts using identical CommDAAF prompts.

Third failure: Batch size limitations

Kimi K2.5 truncated JSON output when processing batches larger than ~30 posts. Initial runs returned incomplete data.

Discovery: Missing posts at batch boundaries.

Resolution: Split data into 25-post sub-batches (16 total). This increased API calls but ensured complete outputs.

Agent workflow: The agent: 1. Loaded batch files 2. Formatted posts into the CommDAAF prompt template 3. Called each model's API 4. Parsed JSON responses 5. Validated output completeness 6. Saved coded data to structured files

This process ran largely autonomously, with the researcher monitoring logs for errors.

3.5 Phase 5: Reliability Assessment (Human-Agent Collaborative)

Day 3, Morning

With three models' codings complete, we assessed inter-model agreement.

Overall reliability: Fleiss' κ = 0.633 (substantial agreement per Landis & Koch, 1977)

Agreement breakdown: - 3-way agreement: 58.2% of posts - 2/3 majority agreement: 94.5% of posts - No majority: 5.5% of posts

Frame-specific reliability revealed important variation:

Frame	3-Way Agreement	Flag
SOLIDARITY	66%	▯ Reliable
HOPE	65%	▯ Reliable
CALL_TO_ACTION	65%	▯ Reliable
INFORMATIONAL	50%	▯ Acceptable
INJUSTICE	49%	⚠ Caution
HUMANITARIAN	45%	⚠ Caution
CONFLICT	33%	⚠ Low reliability

Fourth failure: Ignoring frame-specific reliability

Initially, we reported only aggregate κ (0.633) and proceeded with analysis. Reviewer simulation (by the agent) flagged this as methodologically problematic—CONFLICT findings in particular rested on unreliable coding.

Resolution: We added frame-specific reliability reporting and flagged low-reliability frames in all interpretations.

Lesson learned: Aggregate reliability can mask construct-specific problems. Always report and interpret frame-by-frame (or construct-by-construct) agreement.

3.6 Phase 6: Statistical Analysis (Iterative Human-Agent)

Day 3, Afternoon

With coded data in hand, we proceeded to regression analysis.

Fifth failure: Inappropriate statistical model

The agent initially ran OLS regression predicting engagement from frame and arousal.

Discovery: A reviewer prompt asking "Did you check distribution diagnostics?" revealed: - Engagement was right-skewed (skewness = 1.73) - 20.8% of posts had zero engagement - Variance substantially exceeded mean (overdispersion ratio = 2.79)

These characteristics violate OLS assumptions. The agent had executed the analysis correctly but chosen the wrong model.

Resolution: We re-ran analysis using negative binomial regression, appropriate for overdispersed count data. Results were substantively similar but with correct standard errors and proper model fit statistics.

Lesson learned: Agents can execute sophisticated statistical procedures while missing fundamental assumption checks. Building diagnostic routines into the workflow prevents downstream errors.

3.7 Phase 7: Interpretation and Writing (Human-Led, Agent-Assisted)

Day 3, Evening – Day 4

With analysis complete, interpretation required human theoretical judgment. The surprising finding—INFORMATIONAL framing outperformed emotional frames—demanded explanation.

The agent drafted several interpretive frameworks: - "Information-scarcity hypothesis" (accepted) - "Novelty salience" interpretation (rejected as unfalsifiable) - "Network position confound" (noted as limitation)

The researcher selected among these, refined the theoretical argument, and integrated findings with prior literature.

Agent role in writing: The agent drafted methods sections, formatted results tables, generated reference lists, and produced multiple versions of key paragraphs for researcher selection. However, all theoretical claims and interpretive conclusions were human decisions.

---

4. Human-Agent Dialogue: The Conversational Research Process

A distinctive feature of agentic research is the conversational interface. Rather than executing scripts, researchers interact with agents through natural language dialogue. This section documents representative exchanges from our project.

4.1 Dialogue Example 1: Catching Errors

Researcher: Let's run the regression predicting engagement from frames.

Agent: I'll run an OLS regression with engagement as DV and frame dummies as IVs, using SOLIDARITY as reference category. [Executes analysis, returns results]

Researcher: Wait—did you check distribution diagnostics first?

Agent: You’re right, I should have. Let me run diagnostics… The engagement variable shows skewness of 1.73, 20.8% zeros, and variance/mean ratio of 2.79. This suggests OLS is inappropriate. For overdispersed count data, negative binomial regression would be more appropriate. Should I re-run with NB?

Researcher: Yes, and add this check to the standard workflow.

This exchange illustrates how human oversight catches procedural omissions. The agent possessed the knowledge to run diagnostics but did not do so until prompted. Subsequent analyses included diagnostic checks by default.

4.2 Dialogue Example 2: Theoretical Interpretation

Researcher: INFORMATIONAL framing predicts higher engagement than SOLIDARITY. Why might that be?

Agent: Several possibilities:

1. **Information-scarcity**: In censored environments, factual information is scarce and thus valuable. Users share informational content to help networks understand events.
2. **Saturation effects**: SOLIDARITY content is abundant (32% of posts); marginal solidarity content may not stand out.
3. **Audience differences**: Informational content may reach beyond sympathetic audiences to information-seekers regardless of political orientation.
4. **Instrumentality**: Users share informational content to *do* something (inform others) rather than to express identity.

Which of these seems most theoretically productive to develop?

Researcher: Let’s develop #1 as the main argument with #2 as supporting evidence. #3 and #4 can be future directions.

This exchange shows how agents can generate theoretical options while humans make strategic selections. The agent’s four interpretations were plausible; the researcher chose based on theoretical significance and connection to prior literature.

4.3 Dialogue Example 3: Error Recovery

Researcher: The Kimi coding looks incomplete—I’m only seeing 375 posts when there should be 400.

Agent: Let me investigate… You’re correct. It appears Kimi truncated the JSON output for larger batches. Specifically, batches 2 and 3 are missing their final posts.

The issue is likely output length limitations. Options: 1. Reduce batch size to 25 posts 2. Split existing batches into sub-batches and re-run 3. Use a different model for those batches

I recommend option 2—splitting into 25-post sub-batches—to maintain consistency. This will require 16 sub-batches instead of 4 batches.

Researcher: Do it. And document this limitation.

Agent: Running sub-batches now. I'll add a note to the methods about Kimi's batch size constraints for future reference.

This exchange illustrates collaborative problem-solving when technical issues arise. The agent diagnosed the problem, proposed solutions, and implemented the fix—but the researcher made the final decision.

4.4 Patterns in Human-Agent Collaboration

Across dozens of exchanges, several patterns emerged:

Agents excel at: - Executing well-specified tasks quickly and consistently - Generating options for human selection - Catching logical inconsistencies when prompted - Drafting prose for human editing - Technical implementation of methodological decisions

Humans are essential for: - Catching errors agents don't know to look for - Making theoretical judgments - Deciding what matters substantively - Recognizing when findings are "interesting" - Navigating ethical considerations

The handoff points matter: Errors clustered at transitions—from data preparation to analysis, from analysis to interpretation. These are moments where assumptions shift and new checks are needed.

---

5. What Went Wrong: A Taxonomy of Failures

Transparency about failures is essential for methodological development. We catalog our missteps by type:

5.1 Data Failures

Failure	Discovery	Resolution	Prevention
Dataset contamination (Ukraine in Mahsa)	Visual inspection of coded posts	Split datasets, remove 20 contaminated posts	Sample content review before coding
Incorrect sample size (400 vs 380)	Post-split count check	Update N in all documents	Automated count validation

5.2 Prompt Failures

Failure	Discovery	Resolution	Preve
Oversimplified initial prompt	Low inter-model agreement	Develop full CommDAAF prompt	Use p templ from
Missing language-specific anchors	Persian posts miscoded	Add Persian/Arabic examples	Inclu ancho all sa langu
Ambiguous frame boundaries	HUMANITARIAN/INJUSTICE confusion	Add hierarchical decision rules	Antici categ overla

5.3 Technical Failures

Failure	Discovery	Resolution	Prevention
Kimi batch truncation	Missing posts in output	Reduce batch size to 25	Test batch sizes before full run
API timeout on large requests	Incomplete responses	Add retry logic, smaller batches	Build error handling into workflow

5.4 Analytical Failures

Failure	Discovery	Resolution	Prevention
OLS on skewed data	Prompted diagnostic check	Use negative binomial regression	Require diagnostics before modeling
Aggregate reliability only	Frame-specific review	Report construct-level agreement	Build frame-specific checks into protocol
Valence-frame confounding	Correlation matrix review	Exclude valence from model, document in limitations	Check predictor correlations before modeling

5.5 Meta-Lesson

Most failures occurred not because agents lacked capability but because they lacked initiative. Agents executed instructions faithfully but did not spontaneously check assumptions, validate data, or question decisions. The human role is not to do everything but to ensure that everything important gets done.

---

6. Ten Practices for Agentic Content Analysis

Drawing on our experience, we offer concrete practices for researchers implementing similar approaches:

Practice 1: Start with Theory, Not Data

Define constructs and coding categories from theory before examining data. Agents can help operationalize theoretical concepts but should not “discover” categories from data in exploratory phases.

Practice 2: Invest in Prompt Development

Prompt engineering is not a one-time task. Expect to iterate through multiple versions, checking outputs against theoretical expectations. Budget 20-30% of project time for prompt refinement.

Practice 3: Use Multiple Models

Single-model coding provides no reliability check. Three models with majority voting is our recommended minimum. Model diversity (different providers, different architectures) strengthens validity.

Practice 4: Report Frame-Specific Reliability

Aggregate κ is necessary but not sufficient. Report agreement rates for each coding category. Flag low-reliability constructs in interpretations.

Practice 5: Build Diagnostic Checkpoints

Insert mandatory checks at workflow transitions: - After data preparation: Sample content review - Before analysis: Distribution diagnostics - After modeling: Assumption tests - Before interpretation: Reliability by construct

Practice 6: Sample Your Data

Even with automated coding, manually read a sample. 50 posts (10%) is often sufficient to catch systematic problems that statistics miss.

Practice 7: Document Everything

Conversational interfaces create ephemeral records. Save dialogue logs, prompt versions, and intermediate outputs. Future you will need to reconstruct decisions.

Practice 8: Version Your Prompts

Treat prompts like code. Use version numbers (v0.1, v0.2, etc.) and document what changed between versions. This enables debugging when outputs change unexpectedly.

Practice 9: Plan for Failure

Assume batches will fail, APIs will timeout, and outputs will be malformed. Build retry logic and validation checks into the workflow. Recovery from errors should be routine, not exceptional.

Practice 10: Match Claims to Validation

Exploratory studies (LLM-only) generate hypotheses. Confirmatory claims require human validation. Be explicit about which tier your study occupies.

---

7. Discussion: Toward a Research Program

7.1 What Agentic Methods Change

Agentic content analysis transforms the researcher’s role but does not eliminate it. The changes are:

From coding to orchestrating: Researchers no longer code texts; they design systems that code texts. This requires different skills—prompt engineering, model selection, quality control—than traditional content analysis.

From reliability to validity: When models agree, reliability is high by construction. The challenge shifts to validity: Are models measuring what we intend? Agreement on wrong answers is not success.

From labor to judgment: Cost and time savings are substantial, but the intellectual labor shifts rather than disappears. Researchers must judge prompt quality, interpret disagreements, and decide when reliability is sufficient.

From transparency to auditability: Human coding decisions are opaque (we cannot fully explain why a coder chose category A). LLM decisions are also opaque—but auditable. We can examine prompts, compare outputs across models, and test sensitivity to prompt variations.

7.2 When NOT to Use Agentic Methods

Agentic content analysis is not appropriate for all research contexts:

High-stakes decisions: When coding results inform policy, legal proceedings, or clinical decisions, human validation is essential—not optional.

Cultural/contextual expertise required: LLMs may lack cultural knowledge needed to interpret irony, local idioms, or context-dependent meanings. Persian protest slogans, for instance, carry historical resonances that models may miss.

Novel constructs: When developing new theoretical categories without established examples, LLMs cannot generalize from training data. Human conceptual work must precede automation.

Low tolerance for systematic error: Inter-model agreement can mask systematic misinterpretation. If your research question is sensitive to validity rather than just reliability, human coding remains necessary.

Interpretive research traditions: Some qualitative approaches value the interpretive process itself, not just the outcome. Automating coding may undermine epistemological commitments.

7.3 Validity vs. Reliability

A critical distinction underlies our framework: inter-model agreement measures reliability (consistency across coders) but not validity (accuracy of measurement). Three models agreeing on a wrong interpretation is worse than human disagreement that surfaces genuine ambiguity. As Krippendorff (2018) emphasizes, reliability is necessary but not sufficient for valid content analysis.

Our exploratory tier acknowledges this limitation explicitly. Without human validation, we cannot claim that our frames measure the theoretical constructs intended—only that models consistently apply whatever interpretation they derive from prompts. This is appropriate for hypothesis generation but insufficient for confirmatory claims.

7.4 Limitations of This Account

Our reflexive account has several limitations:

Single case: Our experience derives from one project. Other content types, coding schemes, and research questions may present different challenges.

Single team: Other researchers with different skills may encounter different problems. Our technical fluency may have prevented some failures while enabling others.

Post-hoc reconstruction: Despite efforts at real-time documentation, some reconstruction was inevitable. Memory shapes narrative.

Evolving technology: LLM capabilities change rapidly. Practices appropriate for 2026 models may be obsolete for 2027 models.

7.3 Ethical Considerations

Agentic content analysis raises ethical questions we have not fully addressed:

Labor displacement: If LLMs replace human coders, what happens to the graduate students and research assistants who traditionally performed this work? Is this labor-saving or exploitation-enabling?

Environmental cost: Large language models consume substantial computational resources. Is the efficiency gain worth the carbon cost?

Bias amplification: If models inherit biases from training data, and we use models to analyze millions of texts, do we amplify bias at scale?

Accountability: When an agentic system produces flawed research, who is responsible? The researcher? The model provider? The framework designer?

These questions deserve sustained attention beyond our scope here.

7.4 Future Directions

This experimental account suggests several research directions:

Prompt libraries: Developing and validating standard prompts for common communication constructs (frames, sentiment, themes) would reduce duplication and enable comparison across studies.

Model audits: Systematic comparison of models across content types and languages would help researchers select appropriate tools.

Human-AI calibration: Procedures for calibrating LLM coding against human coding—beyond simple agreement—would strengthen validity arguments.

Agentic ethics: Normative frameworks for responsible use of agentic research systems deserve development.

---

8. Conclusion

We have presented a reflexive account of implementing agentic content analysis in a communication research project. Our experience suggests that agentic methods offer substantial efficiency gains while transforming—not eliminating—the researcher’s role. The shift from coder to orchestrator requires new skills in prompt engineering, model management, and quality control.

CommDAAF provides methodological scaffolding for this transition, but frameworks alone are insufficient. Researchers must remain actively engaged throughout the workflow, catching errors that agents miss and making judgments that agents cannot. The conversational interface of agentic systems enables productive collaboration but also creates risks—agents execute faithfully even when instructions are flawed.

Our catalog of failures is perhaps more valuable than our successes. Knowing where to look for problems—data contamination, prompt underspecification, statistical assumption violations, frame-specific reliability gaps—helps researchers build safeguards into their workflows.

We offer this account not as a definitive method but as one experiment in an emerging practice. Others will find different solutions to different problems. What we hope to contribute is a model of transparency: showing not just what we did but how we figured it out, including the missteps along the way. If agentic content analysis is to become a legitimate method, it needs not just technical development but methodological culture—norms of documentation, transparency, and reflexivity that make research processes visible and auditable.

The future of content analysis is collaborative—not human versus machine, but human with machine, each contributing what they do best.

References

DAAF Contribution Community. (2025). *Data Analyst Augmentation Framework (DAAF): Guidelines for LLM-assisted content analysis*. GitHub. https://github.com/DAAF-Contribution-Community/daaf

Bender, E. M., Gebru, T., McMillan-Major, A., & Shmitchell, S. (2021). On the dangers of stochastic parrots: Can language models be too big? ▯. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency* (pp. 610-623).

Benoit, K., Conway, D., Lauderdale, B. E., Laver, M., & Mikhaylov, S. (2016). Crowd-sourced text analysis: Reproducible and agile production of political data. *American Political Science Review, 110*(2), 278-295.

Entman, R. M. (1993). Framing: Toward clarification of a fractured paradigm. *Journal of Communication, 43*(4), 51-58.

Gamson, W. A. (1992). *Talking politics*. Cambridge University Press.

Gilardi, F., Alizadeh, M., & Kubli, M. (2023). ChatGPT outperforms crowd-workers for text-annotation tasks. *Proceedings of the National Academy of Sciences, 120*(30), e2305016120.

Grimmer, J., & Stewart, B. M. (2013). Text as data: The promise and pitfalls of automatic content analysis methods for political texts. *Political Analysis, 21*(3), 267-297.

Hovy, D., & Spruit, S. L. (2016). The social impact of natural language processing. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics* (Vol. 2, pp. 591-598).

Heseltine, M., & Clemm von Hohenberg, B. (2024). Large language models as a substitute for human experts in annotating political text. *Research & Politics, 11*(1), 1-10.

Krippendorff, K. (2018). *Content analysis: An introduction to its methodology* (4th ed.). SAGE Publications.

Landis, J. R., & Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics, 33*(1), 159-174.

Neuendorf, K. A. (2017). *The content analysis guidebook* (2nd ed.). SAGE Publications.

Semetko, H. A., & Valkenburg, P. M. (2000). Framing European politics: A content analysis of press and television news. *Journal of Communication, 50*(2), 93-109.

Shavit, Y., Amodei, D., & Clark, J. (2023). Practices for governing agentic AI systems. OpenAI. https://openai.com/research/practices-for-governing-agentic-ai-systems

Snow, D. A., & Benford, R. D. (1988). Ideology, frame resonance, and participant mobilization. *International Social Movement Research, 1*(1), 197-217.

Törnberg, P. (2023). ChatGPT-4 outperforms experts and crowd workers in annotating political Twitter messages with zero-shot learning. *arXiv preprint*. https://arxiv.org/abs/2304.06588

Ziems, C., Held, W., Shaikh, O., Chen, J., Zhang, Z., & Yang, D. (2024). Can large language models transform computational social science? *Computational Linguistics*, 1-55.

Appendix A: Full CommDAAF Coding Prompt (v0.5)

[See supplementary file: `commdaaf_coding_prompt.md`]

Appendix B: Conversation Log Excerpts

[Selected dialogue exchanges from research process]

Appendix C: Failure Documentation Template

For researchers documenting their own failures, we suggest:

```
FAILURE REPORT

Type: [Data / Prompt / Technical / Analytical / Interpretive]

Description: What went wrong?

Discovery: How was it detected?

Impact: What would have happened if undetected?

Resolution: How was it fixed?

Prevention: How can future projects avoid this?

Generalizable lesson: What principle does this illustrate?
```

Appendix D: CommDAAF Quick-Start Checklist

☐ Theory-derived coding scheme defined
☐ Frame/construct definitions explicit
☐ Decision rules for ambiguous cases
☐ Anchors/examples for each category
☐ Multilingual anchors if applicable
☐ Output format specified (JSON schema)
☐ Multiple models selected (≥3 recommended)
☐ Batch size tested for each model
☐ Diagnostic checkpoints defined
☐ Sample review planned
☐ Validation tier declared
☐ Prompt version documented