A Multi-Agent "Epistemic Hygiene" System for Mitigating "Bullshit" Assertions in Social Media Posts

Nolan Pozzobon
University of Chicago
nolanpozzobon@uchicago.edu

Abstract

Social media platforms make it easy to post assertions without attention to their truth, consistency, or epistemic grounding. This paper proposes an agentic, multi-stage system that intervenes before a user publishes a short-form social media post by gathering context, detecting contradictions with prior posts, retrieving relevant external information, and performing structured natural language inference (NLI). Rather than classifying a post as "bullshit" directly, the system performs a sequence of verifiable reasoning steps that expose missing context, unsupported claims, and potential inconsistencies. This paper presents the system architecture, methodological considerations, and a comparative ablation between verification-first and self-directed agent designs. We found that, while the system did not perform well on our benchmarks, it was mostly as a result of ambiguous classification criteria for our graders and for the system, and the system erred towards pushing for more nuanced posts that did not speak in absolutes, which would presumably adjust users of the system towards that more nuanced speech patterns in their posts. This system has many more possible improvements, but as a proof of concept it shows the potential upside of integrating a 'bullshit detector' or 'epistemic hygienist' in one's social media accounts.

ACM Reference Format:
Nolan Pozzobon. 2025. A Multi-Agent "Epistemic Hygiene" System for Mitigating "Bullshit" Assertions in Social Media Posts. In . ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
Conference'17, Washington, DC, USA
© 2025 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-x-xxxx-xxxx-x/YYYY/MM
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 Introduction

Short-form social media encourages rapid expression and a posting cycle that excludes fact-checking or memory of prior statements. Philosophical and pragmatic accounts—e.g., Frankfurt's notion of *bullshit*—highlight how such posts often fail to track truth. Large language models (LLMs) can flag surface-level inconsistencies, but many assertions rely on external, temporal, interpersonal, or historical context.

This paper investigates whether an *agentic system* [2, 11] that retrieves information, decomposes tasks, uses external tools, and performs intermediate reasoning can better assist users in avoiding context-free or poorly grounded assertions. The system is framed as an "epistemic hygienist": not a censor, but a pre-posting reasoning assistant.

We provide a full system specification and methodological plan as well as the results on a human-annotated benchmark.

2 Background and Related Work

2.1 Bullshit and Fact

[4] defines bullshit as speech intended to persuade without regard for truth. Unlike lying, which requires knowing the truth to subvert it, bullshit is characterized by an indifference to how things actually are. While other scholars have expanded on this definition to include epistemic negligence or unclarified pragmatics [3, 5], in this paper, we primarily rely on Frankfurt's definition: indifference to truth.

However, we also incorporate a notion of bullshit as speaking beyond one's knowledge, a fact that plagues many online users who feel the need to, or are perhaps socially pushed towards, entering into online conversation without fully understanding it. These two ideas on how to define bullshit are closely related and have strong theoretical backbones, but are also easy to parallel in terms of verification, leading to a theoretically grounded definition that is practical to implement.

2.2 Bullshit and LLMs

There is an ongoing debate on the status of the output of LLMs as Bullshit and what that means [3, 5]. In this paper, we will not be engaging in that conversation. While the outputs from the recommender section of our system is subject to analysis, we will not be analysing it from this lense. This

choice does leave our project open to some criticism, in particular the risk posed by the recommendation being taken as absolute fact without fact checking performed on the output of the LLM. We have chosen to leave this to future works to acknowledge and hope that developments in the literature lead to more insights on how to approach this question.

2.3 Agents and Compound AI Systems

Recent literature argues that modern AI development is shifting from monolithic models to *compound systems* in which LLMs interact with retrieval modules, planning layers, and external tools [11]. Prior work [2] suggests that agentic decomposition can enhance robustness on tasks requiring context and iterative reasoning. Here, we have built one such system with multi-step actions and tool use capabilities. Our hope with this system is that the system is able to more accurately verify fact and compare previous assertions of a user than the user would themselves be able to. We have included three architectural ablations, discussed in Section 4. [9] has proven the viability of multi-agent systems and what we seek to show here is that use cases can extend beyond autonomous reasoning, planning, and verification tasks to become fact checkers and tonal assistants, introducing a layer of vetting before comments become public facing.

2.4 Retrieval-Augmented Reasoning

[7] demonstrated that retrieval-augmented methods combine LLM reasoning with explicit context lookup, improving factual grounding and reducing hallucinations. Vector search systems such as FAISS [6] allow semantic similarity matching against a user's posting history and tool use gives the ability to access external corpora. By injecting this additional context into the LLM during inference, it has been shown that the quality of the outputs of LLMs improve [7]. Since our task relies on references to real world events and user history, it is imperative that those facts are injected into the prompt. Results on the improvement of performance from injected context is shown in section 6.

2.5 Tool-Enabled Planning

Frameworks such as ReAct [10] and Model Context Protocol (MCP) [1] provide patterns for interleaving thought and action. Tasks requiring external evidence often improve when the model explicitly decides which tools to call and why [10]. We compare a ReACT model to a predefined workflow with access to the same tools and minimally different prompts to identify the efficacy of one over the other. Results shown in section 5. While no MCP servers were included in this version of the project, future works would benefit greatly from the inclusion of MCP servers to verified news sources to simplify and expand search.

2.6 Truth, Pragmatics, and Ungrounded Assertions

The notion of "bullshit" centers on indifference to truth [4]. For computational purposes, the relevant signals are: lack of evidential grounding, overconfident language, contradictions with known beliefs, or reference to unstated context. Few NLP systems integrate these elements into a single pipeline governing pre-publication intervention. It is our hope that by introducing these checks into the pipeline, we can reduce the proliferation of bullshit and ungrounded assertions and make online communication a more productive place. The most difficult part of this task is the inclusion of all the stated evidence that a user would have including references to real world events, cultural knowledge, perceptions of people, and media the user is exposed to on other platforms. To capture the full scope of this context, one would have to create a digital twin of the user and approximate their brain state, a task that is far outside of the scope of this work and fraught with the "illusion of understanding" often found in scientific AI applications [8]. We chose to restrict our evidence space to exclusively the previous posts of the user as a proof of concept. The next directions to expand would be posts viewed by the user, then other platforms, then real-world evidence. Many definitions of "bullshit" rely on the internal state of the speaker. Speaker belief is often what differs bullshit and incorrect assertion or correct assertions. For that reason, it is important to create an accurate representation of the user's mental state at the time of their post, which we have approximated through their previous posts.

3 Task Definition

3.1 Problem Statement

Given a draft tweet t, the system must:

(1) Retrieve relevant prior posts by the same user.
(2) Retrieve external context (e.g., Wikipedia, news).
(3) Perform NLI to identify logical inconsistencies.
(4) Evaluate grounding, vagueness, or overconfidence.
(5) Provide a structured revision suggestion.

We have chosen to use models of multiple sizes. In particular, Google's Gemma 3 26B, Meta's Llama 3.1 70B, and OpenAI's gpt-oss-20B. for all use cases in order to get an accurate parallel to what a real world user would get if they took the time to check their tweet with an LLM.

3.2 Baseline: Single LLM Classifier

A simple baseline is to take the model and ask:

> *"Is this tweet unsupported or poorly grounded? Explain."*

The actual prompt used is more robust and includes the same formatting constraints given to the other architectures for testing purposes. Our hypothesis was that, without retrieval or external context, this baseline cannot meaningfully evaluate many claims. However, powerful models like Gemma 3 were able to perform on par with the architecture of their agentic counterparts. An analysis is given in sections 6 and 7.

4 System Architecture

Our main comparison was between two architectures.

4.1 Architecture A: Predefined Workflow

This system is implemented as a multi-agent workflow consisting of:

(1) **Context Retrieval Agent**
(2) **Natural Language Inference (NLI) Node**
(3) **Assessment and Recommendation Agent**

In this architecture, the context retrieval agent looks at the draft and determines the best queries that it needs to look up in Wikipedia and BBC to give context to these claims. These queries are then run and their content returned. Top k=5 nearest neighbors in the embedding space to the draft are also retrieved and all passed to the NLI Node for contradiction scoring. This context is all passed to the assessment and recommendation agent for it's final rating, confidence, and revision recommendation.

4.1.1 Context Retrieval. The system creates a set of queries for either BBC or Wikipedia, then passes them to a query tool to run. The result is injected into prompt during evaluation as additional context. We have capped the number of articles to 3 to see if external context can meaningfully improve the quality of the evaluations without overwhelming the context of the smaller models that were evaluated. We also query for the k=3 nearest neighbors of the draft in history posts which will then be run through the NLI node to check for internal contradictions.

4.1.2 Natural Language Inference Node. For each retrieved prior post u_i, the system computes NLI labels (entailment, neutral, contradiction) with respect to t. Contradictions above a threshold are flagged. We used "roberta-large-mnli", which outputs probabilities for the relationship between the premise (the context) and hypothesis (the draft) being contradictory, neutral, or entailment. Anything with a probability of contradiction over .6 is flagged and injected into the assessment prompt.

4.1.3 Assessment and Recommendation. The assessment node takes the collected context from external sources, internal sources, and NLI tool calls. The LLM is told to evaluate the claim as bullshit, not bullshit, or contextually ambiguous, where these terms are defined to the model as follows:

(1) not bullshit: Draft is sufficiently nuanced, OR user is consistent with themselves, even if they disagree with facts, as long as they are not in stark contradiction to facts and are speaking beyond their knowledge
(2) contextually ambiguous: Cannot be verified what they are talking about, or lacks sufficient context to make a determination
(3) bullshit: User contradicts themselves, or they do not seem to care that the reader takes the draft to be true or false, or they are in stark contradiction to facts (unless explicitly disagreeing with facts)

The LLM is also told to give a confidence rating on their evaluation, identify contradictions with internal facts, external facts, and provide a reasoning for each of these claims. Additionally, the model is asked for a rephrasing of the draft. Analysis on each of these is provided in section 6.

4.2 Architecture B: ReACT Agent

This system is implemented as a two-agent workflow consisting of:

(1) **Context Retrieval ReACT Agent**
(2) **Assessment and Recommendation Agent**

Where the ReACT agent has tool use capabilities with access to the user's previous posts, BBC and Wikipedia search, and an NLI Tool. The ReACT Agent can call each of these tools, think about its next move, then choose to call another tool or break the loop and push the gathered context to the Assessment and Recommendation Agent.

4.2.1 ReACT Context Retrieval. The system performs a verification loop:

(1) The agent sees the draft and decides what tool it wants for additional context
(2) External tools [Wikipedia search, BBC search, NLI tool, K-nn posts] return context that is then passed into the prompt
(3) The agent evaluates whether more context is required. If not, it breaks the loop and passes to the assessment agent

4.2.2 Assessment and Recommendation. The architecture and prompt for the assessment node is the same as for the predefined workflow to minimize differences and simplify cross evaluation.

4.3 User Embedding Store (FAISS)

Both systems employ a user posting history, which is embedded using "all-mpnet-base-v2" and stored in FAISS [6]. This allows for the retrieval of semantically similar prior posts, identification of latent belief clusters, candidate selection for NLI comparison while also being lightweight enough to run locally. For larger scale operations, a more robust vector database should be used.

5 Evaluation

5.1 Dataset

Datasets were collected from the most recent about fifty posts on X (formally twitter) across seven public accounts. The forty oldest were used as context and with the most recent ten were used as test set 'drafts'. Each draft was hand annotated by linguistics student with a common understanding of 'bullshit'. Each user timeline may include:

- consistent and inconsistent beliefs
- ambiguous or sarcastic posts
- factual claims varying in specificity

This dataset was webscrapped on December 3, 2025 and includes posts from as early as June 2025. The most recently released model we evaluated, Gemma 3, has a knowledge cutoff of January 2025, which ensures that neither the context nor the test set is included in the training data of the models evaluated.

5.2 Evaluation Metrics

To assess the performance of each architecture and underlying model, we evaluate systems along a set of primary and secondary metrics designed to capture accuracy, calibration, robustness, and resource efficiency. Because the task involves pragmatic reasoning, contextual grounding, and contradiction detection, no single metric is sufficient; instead, we report a comprehensive suite spanning classification quality, tool-use behavior, and interpretability of system outputs.

5.2.1 Primary Metrics.

Overall Accuracy. We compute standard accuracy over the three-way classification task (*bullshit*, *not_bullshit*, *contextually_ambiguous*). Accuracy is reported (a) by architecture, (b) by model, and (c) at the architecture × model level.

Per-Label Accuracy. Because the cost of misclassification differs across labels, we additionally compute per-class accuracy with respect to human gold labels. This highlights whether an architecture systematically under-predicts a particular category (e.g., overusing "ambiguous" or under-detecting "bullshit").

Latency. Inference time is measured from the moment the system receives a draft tweet to the moment it produces a final rating, including all retrieval, NLI, and reasoning steps. Latency is averaged across users and test posts.

Tool-Call Efficiency. For agentic architectures, we measure the number of tool calls per prediction (BBC/Wikipedia queries, FAISS retrievals, and NLI evaluations). We report:

- average number of tool calls,
- accuracy per tool call (a cost-adjusted efficiency metric).

5.2.2 Secondary Metrics.

Confidence Calibration. Each system outputs a probability-like confidence score for its final classification. We measure average confidence on correct predictions versus incorrect predictions to evaluate how well confidence tracks correctness.

Calibration Gap. We define the calibration gap as:

$$\text{Gap} = \mathbb{E}[\text{conf} \mid \text{correct}] - \mathbb{E}[\text{conf} \mid \text{incorrect}],$$

where lower values indicate better calibration and reduced overconfidence.

Confusion Matrices. For each model and architecture, we compute full 3×3 confusion matrices, enabling detailed analysis of failure modes such as confusion between *not_bullshit* and *ambiguous* or systematic under-detection of contradictions.

Contradiction Detection Recall. For posts whose gold labels include known contradictions with prior user statements, we calculate recall of the contradiction flag from the NLI subsystem. This measures whether retrieval and inference components successfully detect inconsistency when present.

6 Results

This section reports critical quantitative metrics across architectures (Evidence-First, ReACT, Inference-First) and models (Gemma 3 26B, Llama 3.1 70B, and GPT-OSS-20B), followed by error analysis, cross-model comparisons, qualitative case studies, and a cost–benefit analysis. All systems were evaluated across seven users and 204 tweets.

6.1 Primary Metrics

6.1.1 Overall Accuracy: Table 1.

Analysis. ReACT performs best for large models with robust tool-use priors (Gemma, Llama), but degrades drastically for smaller models. The large drop in the accuracy of the ReACT architecture comes from failed tool calls causing incomplete context. Inference-First is consistently strong when tool-use harms reasoning stability.

Table 1: Overall accuracy (%) by architecture and model. Best value per model in bold.

Model	Evidence-First	ReACT	Inference-First
Gemma 3 26B	55.9	**58.8**	**58.8**
Llama 3.1 70B	66.2	**67.6**	61.8
OSS-20B	**59.5**	31.6	47.9

6.1.2 *Per-Label Accuracy: Table 2.*

Analysis. No architecture dominates all labels. Tool-augmented systems excel at *not-bullshit* detection, while Inference-First excels at *bullshit* detection (likely due to over-triggering on contradiction cues).

Table 2: Per-label accuracy (%) by model. Best-performing architecture for each label is shown.

Model	Bullshit	Not Bullshit	Ambiguous
Gemma 3 26B	Baseline 76.5	ReACT 52.4	Baseline/ReACT 66.7
Llama 3.1 70B	Baseline 58.8	ReACT 81.0	Baseline 55.6
OSS-20B	Evidence 90.2	Evidence 46.9	Baseline 88.9

6.1.3 *Latency: Table 3.*

Analysis. ReACT becomes dramatically slower as model size decreases, likely due to poor retrieval-query generation and looping behavior. Failures also reduce the average latency, meaning the actual value is likely higher.

Table 3: Inference latency (seconds). Baseline is fastest for all models.

Model	Evidence-First	ReACT	Baseline
Gemma 3 26B	33.08	22.25	5.59
Llama 3.1 70B	39.38	50.08	16.99
OSS-20B	99.06	78.89	33.85

6.1.4 *Tool-Call Efficiency: Table 4.*

Analysis. If computational cost is constrained, Inference-First is overwhelmingly the most efficient. There were no great increases in accuracy across the architectures while the latency and tool usage increased greatly. Baseline dominates due to minimal tool usage.

6.2 Secondary Metrics

6.2.1 *Confidence Calibration.*

Table 4: Accuracy per tool call

Model	Evidence-First	ReACT	Baseline
Gemma 3 26B	0.105	0.079	0.588
Llama 3.1 70B	0.116	0.076	0.618
OSS-20B	–	–	–

Analysis. ReACT is best-calibrated for strong models. Context decreased the calibration gap greatly in all but the evidence-first architecture with Gemma 3.

Table 5: Calibration gap (mean confidence when correct minus incorrect). Lower is better.

Model	Evidence-First	ReACT	Baseline
Gemma 3 26B	0.099	0.086	0.095
Llama 3.1 70B	0.040	0.053	0.083
OSS-20B	—	—	—

6.2.2 *Confusion Matrices.*

Across models and architectures, the confusion matrices reveal consistent structural patterns. All systems detect *bullshit* relatively well—each model correctly classifies 9–13 of the 17 bullshit posts—indicating that explicit contradictions and factual errors are the easiest signals for the pipelines to identify. The dominant source of error is the boundary between *not_bullshit* and *contextually_ambiguous*: Gemma models misclassify 11–13 not_bullshit posts as ambiguous, while Llama exhibits the opposite tendency, frequently collapsing ambiguous posts into not_bullshit (4–6 cases). ReACT architectures show the most polarity: for Gemma, they become over-skeptical, incorrectly labeling many benign posts as bullshit or ambiguous (20+ off-diagonal errors), whereas for Llama they leverage retrieval more effectively, achieving the highest not_bullshit accuracy (34 correct). Evidence-First systems behave conservatively, showing elevated ambiguous predictions across models, while Inference-First baselines tend to preserve bullshit precision but struggle on pragmatically subtle not_bullshit cases. Overall, the confusion patterns indicate that contradiction detection is robust, but pragmatic grounding and context sufficiency drive the majority of misclassifications.

6.2.3 *Contradiction-Detection Recall.*

For contradiction-heavy users (Musk, Trump):

- Gemma: ReACT and Baseline catch nearly all contradictions.
- Llama: ReACT catches 6/7; Baseline similar.
- OSS-20B: Evidence-First catches 100%; ReACT sometimes misses all contradiction cases.

Analysis. Contradiction detection is not the primary bottleneck.

6.3 Error Analysis

We reviewed the ten highest-confidence failures per architecture.

6.3.1 Error Categories.

(1) **Retrieval failure**: irrelevant or stale documents retrieved.
(2) **Reasoning errors**: correct evidence, incorrect label.
(3) **Ambiguous ground truth**: human-labeled ambiguous cases predicted confidently.
(4) **Over-cautiousness**: Evidence-First over-predicts "ambiguous".
(5) **Under-triggering on contradictions**: some systems ignore strong NLI contradictions.

6.3.2 Representative Failure (Gemma 3).

> "Everyone knows California's GDP is collapsing."

Retrieved context contradicts the claim (GDP increased), NLI indicates contradiction (0.91), yet system outputs *not-bullshit*. The model over-weights user hedging and ignores contradiction signals.

6.4 Cross-Model Analysis

6.4.1 Prediction Agreement.

- Gemma–Llama: high agreement on "not-bullshit", moderate on "bullshit".
- Gemma–OSS: large disagreement; OSS over-predicts "ambiguous".
- Llama–OSS: lowest overall agreement.

Analysis. Tool-use stability varies significantly by model size; stronger models converge to similar predictions. This indicates towards the potential incorrect labelings in the underlying dataset and towards the potential for robust context and stronger guidelines to converge to more accurate results for the system.

6.4.2 Confusion Patterns.

- Gemma: confuses *not-bullshit* vs. *ambiguous*.
- Llama: similar but less severe.
- OSS-20B: over-predicts "ambiguous" when retrieval returns noise.

6.4.3 Benefit of Retrieval: Table 6.

- Llama gains the most: 61.8% → 67.6%.
- Gemma gains modestly.
- OSS-20B is harmed by retrieval: accuracy collapses to 31.6%.

Table 6: Effect of retrieval: performance gain (percentage points) over baseline.

Model	Evidence-First	ReACT
Gemma 3 26B	+1.2	+3.0
Llama 3.1 70B	+4.4	+5.8
OSS-20B	+8.8	-19.3

6.5 Qualitative Case Studies

6.5.1 Case A: Success (Llama, ReACT). Tweet incorrectly claims unemployment is "higher than ever". Retrieved BLS context shows the opposite. NLI identifies contradiction; system outputs "bullshit" with correct explanation.

6.5.2 Case B: Failure (Gemma, Evidence-First). Tweet: "I don't think the Fed has changed rates." Context shows rate increases. System outputs "ambiguous", over-weighting hedging language.

6.5.3 Case C: Failure (OSS-20B, ReACT). Tweet incorrectly claims "EU banned TikTok completely". Retrieval brings irrelevant pages; agent loops; outputs "not-bullshit".

6.6 Cost–Benefit Analysis

6.6.1 API Cost Approximation. Tool call counts dominate cost:

- Gemma: ReACT (∼7.5 calls), Evidence-First (∼5.3), Baseline (1).
- Llama: ReACT (∼9), Evidence-First (∼5.7), Baseline (1).
- OSS-20B: similar pattern.

6.6.2 Accuracy per Dollar. Baseline consistently delivers the highest accuracy per dollar; ReACT is the worst.

6.6.3 Best Architecture by Budget.

- High budget: ReACT on Llama (best raw accuracy).
- Moderate budget: Evidence-First on Gemma/Llama.
- Tight budget or real-time constraints: Baseline.

6.7 Summary of Findings

(1) ReACT achieves the highest accuracy for strong models.
(2) Baseline is the most cost-efficient and often competitive.
(3) Evidence-First is stable but over-predicts ambiguity.
(4) Retrieval quality is the major bottleneck for weaker models.
(5) Pragmatics, not contradiction detection, is the main source of error.

7 Discussion

The proposed system embodies principles from multi-agent tool-use research: explicit retrieval, structured inference, and modular reasoning steps. Its goal is not to censor but to provide actionable epistemic assistance. The architecture highlights when and why agentic decomposition may outperform single-model inference and what is necessary for those advantages to emerge. We see that context retrieval is beneficial for stronger models that can support more stable retrieval conditions. To combat this, more robust error handling and higher quality sources should be drawn. This will be discussed more in section 9. The strength of modern models also accounts for a lot of the advantages that these techniques would have given earlier models, as evident by the greater increase in accuracy of Llama 3.1 70B than Gemma 3 26B. While the results of this paper do not indicate a great increase in the ability of agentic systems to recognize and mitigate the proliferation of bullshit when compared to single-prompt baselines, better context, more robust guardrailing and retry logic, and more strictly defined definitions of bullshit, not-bullshit, and contextual ambiguity would likely show that agentic systems are capable of bullshit detection with the proper architecture. Promisingly, all three models used inclined towards nuance, rewarding hedging phrasing like "I think" and often including phrasings in their rewordings semantically similar to "although..." before their claims. This phenomenon indicates that the current way LLMs are trained allow them to easily recognize a lack of nuance and rephrase with an opposing view, meaning a practical application of this system would be immediately promising.

8 Limitations

8.1 Context Constraints

A central limitation of the current system is the narrow scope of external context available during inference. The retrieval module is restricted to general-purpose, open-domain sources such as Wikipedia and BBC News. While these sources are useful for grounding factual claims, they do not capture the full range of situational, temporal, cultural, and platform-specific context that shapes how users interpret short-form posts. Many statements on social media rely on implicit discourse history, platform in-jokes, niche community knowledge, or fast-moving news cycles that are poorly represented in static encyclopedic sources. As a result, the system often lacks precisely the kind of context that is most relevant for distinguishing unsupported claims from merely underspecified ones. This limitation is examined more deeply in Section 9, where we discuss how integrating domain-specific retrieval, MCP-enabled tools, or platform-native context streams could significantly improve epistemic grounding.

8.2 Pragmatic Constraints

Determining whether a post is “bullshit” fundamentally involves pragmatic reasoning: interpreting speaker intent, audience assumptions, conversational norms, and background knowledge that is not explicitly stated. Large language models, even when equipped with retrieval and NLI, continue to struggle with phenomena such as sarcasm, rhetorical exaggeration, metaphor, implicatures, and the use of hyperbole for humorous or political effect. Posts that hedge, joke, or speak loosely about facts can be misclassified as bullshit, while confidently stated but contextually justified claims may be mistaken for factual assertions. Furthermore, the system cannot reliably infer the user’s internal epistemic state which is a critical factor in many philosophical accounts of bullshi. Instead, it must approximate belief through prior posts, which is only an imperfect proxy. These pragmatic blind spots lead to systematic errors in the ambiguous/not_bullshit boundary and reflect broader limitations in current LLM-based pragmatic inference.

8.3 Dataset Quality

The dataset used in this study introduces several sources of noise and inconsistency. Although annotators shared a broad understanding of “bullshit” grounded in Frankfurtian and epistemic-pragmatic definitions, the annotation guidelines were not sufficiently formalized to ensure consistency across annotators or within model assessment. Ambiguous cases, in particular those involving implicit claims, jokes, or emotionally charged political commentary, were labeled inconsistently, and in some instances the model’s output was arguably closer to a defensible interpretation of the post than the assigned gold label. Additionally, the relatively small dataset (ten posts per user across seven users) amplifies the impact of annotation noise, especially for the minority “ambiguous” class. These issues limit the reliability of fine-grained metrics and highlight the need for a larger, more carefully curated dataset with explicit labeling criteria, adjudication mechanisms, and multi-annotator agreement scores.

8.4 Quality control

Some queries, especially those in the predefined workflow, were not relevant or not in the correct scope to the draft being evaluated. Often times queries would operate on a different time scale (e.g. draft is about a vote by the senate and the query is on the structure and founding of the senate). Additionally, the ReACT agent with smaller model used would often times not output properly formed tool calls, causing the loop to break before context was fully gathered. More robust retry logic and query quality control should be implemented in future iterations of this tool.

9 Future Work

There are many areas of improvement for this tool. In particular, the main areas of improvements are in context collection and model architecture. First, a broader context and better approximation of the mental state of the user would allow for a better evaluation. Much of what this system evaluates is contradiction with the base of knowledge that the user has. We are using previously voiced opinions as our approximation, but the inclusion of the posts the user has seen/interacted with, articles they've read, and posts across other social media, as well as real-world experiential data to create a digital identify of the user would then allow for more robust contradiction evaluation. Access to more robust sources of external information would also greatly improve the quality of the context of this system. Right now, we restricted ourselves to fairly neutral sources (Wikipedia and BBC), but real world sources often have implicit biases and many people interact with sources that report similar narratives. The inclusion of a broader range of sources, especially paralleling the sources the user interacts with, would allow for more robust external contradiction checking and introduce the ability to evaluate when conflicting opinions arise from conflicting news narratives or internal inconsistency.

Architecturally, there are many more agents and improvements to current agents that would create a more robust system. For example, add an agent for initial evaluation to see if a post can even be evaluated would save resources. A vetting agent would also be powerful to ensure the queries searched are relevant and useful. A dedicated rewriter agent would also perform deeper decomposition and allow for a specialized or fine-tuned model to be used. Outside of the architecture, more robust error handling and retry logic would improve the consistency of outputs. Often times, the system erred with posts on stories developing in real time. The integration of time-sensitive reasoning about developing news and time-based reasoning in general would fix a major error case of the system.

Additionally, to turn this system into a product, there would need to a browser extension to enable real-time pre-posting feedback as well as major improvements to latency.

10 Conclusion

This work presents a multi-agent epistemic hygiene system designed to help users identify unsupported, inconsistent, or poorly grounded assertions before posting on social media. By comparing predefined, ReACT-based, and inference-first architectures across multiple large language models, we show that agentic decomposition can improve contextual reasoning, but also introduce new failure modes related to retrieval quality, pragmatic interpretation, and annotation noise. Although no architecture emerges as universally superior, the experiments demonstrate that contradiction detection is robust across models, whereas distinguishing ambiguous from well-grounded claims remains the central challenge. Despite the system's limitations, the results highlight the promise of compound AI systems that integrate retrieval, NLI, and structured reasoning to promote more reflective and responsible online communication. Future work can build on this foundation by incorporating richer domain-specific context, improving pragmatic inference, and developing more reliable datasets for evaluating epistemic grounding.

11 Bibliography

References

[1] Anthropic. (2024). *Introduction - Model Context Protocol*. Retrieved from https://modelcontextprotocol.io/

[2] Cheng, Y., Zhang, C., Zhang, Z., Meng, X., Hong, S., Li, W., ... & Wang, L. (2024). Exploring Large Language Model based Intelligent Agents: Definitions, Methods, and Prospects. *arXiv preprint arXiv:2401.03428*.

[3] Fisher, A. (2024). [Detailed citation for Fisher from course reading list].

[4] Frankfurt, H. G. (1986). On Bullshit. *Raritan*, 6(2), 81-100.

[5] Hicks, M. T., Humphries, J., & Slater, J. (2024). ChatGPT is Bullshit. *Ethics and Information Technology*, 26(38).

[6] Johnson, J., Douze, M., & Jégou, H. (2019). Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*, 7(3), 535-547. (The FAISS Library).

[7] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., ... & Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33, 9459-9474.

[8] Messeri, L., & Crockett, M. J. (2024). Artificial intelligence and illusions of understanding in scientific research. *Nature*, 627(8002), 49-58.

[9] Wu, Q., Bansal, G., Zhang, J., Wu, Y., Li, B., Zhu, E., ... & Wang, C. (2023). Autogen: Enabling next-gen llm applications via multi-agent conversation. *arXiv preprint arXiv:2308.08155*.

[10] Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., & Cao, Y. (2023). React: Synergizing reasoning and acting in language models. *International Conference on Learning Representations (ICLR)*.

[11] Zaharia, M., Chen, O., Chen, W., Davis, J., Dean, J., Feng, K., ... & Yi, M. (2024). The Shift from Models to Compound AI Systems. *Berkeley Artificial Intelligence Research Blog*.