Extended Abstract: CensorAlert – Leveraging LLM Agents for Automated Censorship Report Aggregation and Analysis

Ali Zohaib
University of Massachusetts Amherst

Jade Sheffey
University of Massachusetts Amherst

Mingshi Wu
GFW Report

Amir Houmansadr
University of Massachusetts Amherst

Abstract

Internet censorship reporting is fragmented across many channels, from measurement platforms to multilingual, crowdsourced reports circulating in chat groups, forums, and on social platforms, making it hard for researchers and advocates to reliably track and act on emerging incidents. In this work, we introduce CensorAlert, a platform that addresses this challenge by aggregating reports from diverse sources and using LLM-based agents to continuously monitor, normalize, translate, and summarize them into a unified format. Each report is scored by an LLM agent for significance and surfaced as part of a ranked feed published at https://censoralert.org. Users can also subscribe to receive timely alerts via email or a supported messaging platform.

1 Introduction

Censorship incidents surface online through two primary channels: systematic measurement platforms and crowdsourced reports. Over the past decade, the Internet freedom community has developed and maintained several large-scale measurement infrastructures, such as Open Observatory of Network Interference (OONI) [4], Censored Planet [1], Cloudflare Radar [3], and NetBlocks [7], that continuously monitor network behavior for anomalies indicative of interference or blocking. These platforms have been instrumental in uncovering both ongoing and historical censorship events, including nationwide Internet shutdowns, protocol-specific disruptions, and other blocking incidents. However, their visibility is inherently limited by the operational constraints of geographic coverage and protocol-specific tests. Consequently, many localized, niche, short-lived, or rapidly evolving forms of censorship remain difficult to detect using deterministic measurement alone [2, 8, 9, 12–15].

In these gaps, users often provide the most critical evidence. People encountering blocked websites, failing messaging apps, or large-scale connectivity issues share observations across diverse channels, languages, and platforms. Community spaces such as Net4People BBS [6] and the NTC Party forum [10] enable users to discuss and describe suspected blocking and offer researchers timely leads. Yet this reporting ecosystem is highly fragmented: relevant information may appear in multiple languages, in closed groups, or in technical forums that are not routinely monitored. Several important cases have only come to light because individual

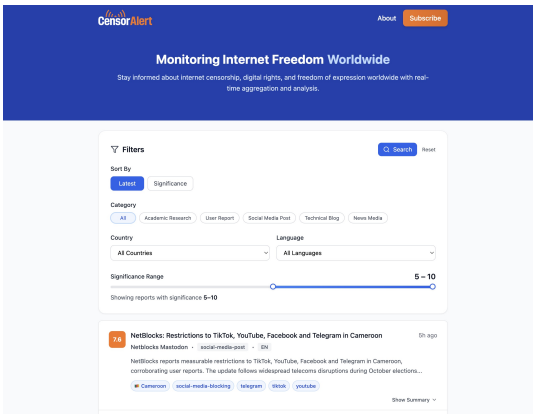

Figure 1: CensorAlert's web interface, showing the most recent and significant posts. Posts are aggregated, summarized, and scored by an LLM-based agent.

experts happened to notice such reports, for example, investigations into blocking of fully encrypted protocols in China were prompted by user reports on an online forum [12], and regional blocking in parts of China has been identified through GitHub issue pages of popular circumvention tools [13]. This informal, crowdsourced pipeline is fragile, relying on whether reports reach a small group of overburdened volunteers and researchers. While the inherently heterogeneous nature of crowdsourced censorship reporting is unlikely to change, there is a clear need to automate the process of collecting and sifting relevant material.

To address this challenge, in this work, we present CensorAlert[1], a platform designed to automate the discovery and assessment of Internet censorship reports. CensorAlert *aggregates* signals from diverse open sources, including multilingual social media posts, measurement platform reports, news articles, technical forums, and blogs, and converts these heterogeneous items into a common schema. The platform then employs large language model (LLM) agents to: (1) extract key attributes (e.g., report type, affected country, censorship mechanism, and impacted services); (2) translate and summarize findings; and (3) score each item's significance based on its impact and strength of evidence. By consolidating noisy, distributed signals into a transparent, prioritized list of censorship-related stories, CensorAlert reduces the monitoring and manual

[1] https://censoralert.org

Free and Open Communications on the Internet 2026(1), 11–13

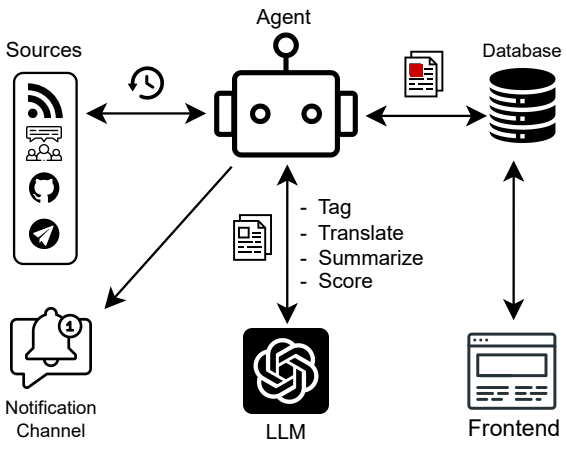

Figure 2: CensorAlert architecture: The agent periodically collects data from defined sources and sends new posts to the LLM for tagging, translation, summarization, and scoring. Posts with high relevance scores are stored in the database, which is then queried by the frontend, and relevant updates are delivered to users via email or message notifications.

filtering burden for volunteers and researchers while simultaneously providing the public with a single, accessible source of timely, well-structured information about global censorship activity.

2 System Design

CensorAlert is a modular system built on N8N [5], an open-source, self-hosted workflow automation platform. This design allows for the flexible composition and portability of automation pipelines. Figure 2 shows an overview of CensorAlert's architecture.

Workflows: N8N Workflows are the core operational units of CensorAlert. Each workflow is a series of connected nodes that performs a discrete task, and in CensorAlert, it is structured around three main components: a data source, an AI agent (with LLM API access), and a data store. These workflows are triggered on a timed interval (e.g., every two hours) and automate the entire data-processing logic: fetching data, calling the AI agent for analysis, parsing the results, writing to the database, and sending alerts. We instantiate distinct workflows from a main template to handle different data source types.

Data Sources: We aggregate data from heterogeneous, openly accessible channels, including but not limited to measurement platform APIs (e.g., OONI Reports, Cloudflare Radar, NetBlocks), GitHub Issues from circumvention tools (e.g., Hysteria, Xray), discussion forum RSS feeds (e.g., Net4People, NTC Party), relevant social media (e.g., Mastodon, X, Telegram channels), and research papers from arXiv. Upon ingestion, each item is normalized into a common internal schema that captures its timestamp, source, raw data, and provenance, while preserving a link to the original content for auditability. While this pipeline is automated, extending the system with new data sources is a simple, though manual, process. We are actively expanding this source list and welcome community contributions.

AI-based Scoring, Clustering, and Deduplication: Data collected from our sources is passed to an LLM-based AI agent for analysis, as manually filtering the large volume of noisy, multilingual data is infeasible. LLMs are well-suited to this setting, as they can effectively read unstructured text, handle multiple languages, and interpret informal or incomplete descriptions.

For each data item, the agent, guided by a system prompt, calls the LLM API to perform several tasks: (i) classify if it is related to Internet censorship, (ii) assign scores between 0-5 for credibility, novelty, impact, timeliness, and verifiability, providing reasoning for each, and (iii) rewrite its title, provide an English summary, and add relevant tags. The LLM is constrained to return this output in a structured JSON format. The agent then calculates a normalized significance score (0-10) from the component scores and adds all censorship-related posts to the PostgreSQL database.

Following this step, the agent generates text embeddings (from the summary, title, and tags) using an embedding model (OpenAI's text-embedding-3-small). These embeddings are used to cluster semantically similar items within specific time windows. Posts with overlapping attributes are grouped, while near-duplicates (such as reposts, copied headlines, or translations) are collapsed into a single canonical post, preserving all original source URLs and metadata.

CensorAlert currently uses OpenAI's GPT-5 Thinking model (hosted on Azure) as the evaluator, chosen for its strong multilingual summarization and classification capabilities [11]. Its behavior is tightly constrained via the prompt, specific examples, and a strict JSON output schema, making the evaluation transparent.

Website Frontend: The CensorAlert frontend, built using React, queries the PostgreSQL database through a public REST API to display a ranked list of high-significance posts. As shown in Figure 1, the interface displays posts with their summaries, significance scores, and key attributes (e.g., country, sources, tags). Users can filter this data by various criteria, including country, time window, and score, or search by keyword.

Notification Channels: Notifications are implemented as an additional scheduled workflow within the same architecture. Users subscribed to the mailing list or Telegram channel receive alerts when new posts exceed configurable significance thresholds or match selected filters. We are actively working on making the notification API publicly accessible and extending support to additional notification channels.

3 Discussion and Future Work

The Internet freedom community relies heavily on the diligent, manual efforts of volunteers. Our goal with CensorAlert is to complement and scale these efforts. By systematically aggregating and organizing censorship-related content, CensorAlert will serve as a key resource, enabling advocates and researchers to stay informed of the latest developments and helping the broader community track censorship activities. In the long term, by leveraging the growing capabilities of AI agents, our aim is to automate the full lifecycle of a censorship event, from initial reporting and detection to measurement and the development of circumvention strategies, with CensorAlert serving as the first step toward this goal.

Acknowledgments

The work was supported in part by the NSF grant CNS-2333965, and by the Young Faculty Award program of the Defense Advanced Research Projects Agency (DARPA) under the grant DARPA-RA-21-03-09-YFA9-FP003.

References

[1] Censored Planet. 2026. Censored Planet. https://censoredplanet.org/.
[2] Zimo Chai, Amirhossein Ghafari, and Amir Houmansadr. 2019. On the Importance of Encrypted-SNI (ESNI) to Censorship Circumvention. In *Free and Open Communications on the Internet*. USENIX. https://www.usenix.org/system/files/foci19-paper_chai_update.pdf
[3] Cloudflare. 2026. Cloudflare Radar. https://radar.cloudflare.com/.
[4] Arturo Filastò and Jacob Appelbaum. 2012. OONI: Open Observatory of Network Interference. In *Free and Open Communications on the Internet*. USENIX. https://www.usenix.org/system/files/conference/foci12/foci12-final12.pdf
[5] n8n. 2025. AI Workflow Automation Platform & Tools - n8n. https://n8n.io/. Accessed: 2025-11-06.
[6] Net4People. 2026. Net4People BBS Issues. https://github.com/net4people/bbs/issues
[7] NetBlocks. 2026. NetBlocks. https://netblocks.org/reports.
[8] Niklas Niere, Felix Lange, Robert Merget, and Juraj Somorovsky. 2025. Transport Layer Obscurity: Circumventing SNI Censorship on the TLS Layer. In *Symposium on Security & Privacy*. IEEE. https://www.computer.org/csdl/pds/api/csdl/proceedings/download-article/26hiUekZ19S/pdf
[9] Sadia Nourin, Erik Rye, Kevin Bock, Nguyen Phong Hoang, and Dave Levin. 2025. Is Nobody There? Good! Globally Measuring Connection Tampering without Responsive Endhosts. In *Symposium on Security & Privacy*. IEEE. https://www.computer.org/csdl/pds/api/csdl/proceedings/download-article/26hiUgw654A/pdf
[10] NTC Community. 2026. NTC Party: "No Thought is a Crime" — Internet Censorship Circumvention Forum. https://ntc.party/
[11] OpenAI. 2025. Introducing GPT-5. https://openai.com/index/introducing-gpt-5/ Accessed: 2025-11-07.
[12] Mingshi Wu, Jackson Sippe, Danesh Sivakumar, Jack Burg, Peter Anderson, Xiaokang Wang, Kevin Bock, Amir Houmansadr, Dave Levin, and Eric Wustrow. 2023. How the Great Firewall of China Detects and Blocks Fully Encrypted Traffic. In *USENIX Security Symposium*. USENIX. https://www.usenix.org/system/files/sec23fall-prepub-234-wu-mingshi.pdf
[13] Mingshi Wu, Ali Zohaib, Zakir Durumeric, Amir Houmansadr, and Eric Wustrow. 2025. A Wall Behind A Wall: Emerging Regional Censorship in China. In *Symposium on Security & Privacy*. IEEE. https://gfw.report/publications/sp25/data/paper/paper.pdf
[14] Diwen Xue, Benjamin Mixon-Baca, ValdikSS, Anna Ablove, Beau Kujath, Jedidiah R. Crandall, and Roya Ensafi. 2022. TSPU: Russia's Decentralized Censorship System. In *Internet Measurement Conference*. ACM. https://dl.acm.org/doi/pdf/10.1145/3517745.3561461
[15] Ali Zohaib, Qiang Zao, Jackson Sippe, Abdulrahman Alaraj, Amir Houmansadr, Zakir Durumeric, and Eric Wustrow. 2025. Exposing and Circumventing SNI-based QUIC Censorship of the Great Firewall of China. In *USENIX Security Symposium*. USENIX. https://gfw.report/publications/usenixsecurity25/data/paper/quic-sni.pdf