good! 25/25

SocialMediaAvengers Project 3 Proposal

Karthik Maganahalli Prakash
Binghamton University
New York, USA
kmaganahalli@binghamton.edu

Sanjana Shivanand
Binghamton University
New York, USA
sshivanand@binghamton.edu

Abstract

In Project 2, we extended the SocialMediaAvengers data pipeline by moving from continuous collection to structured measurement across Reddit and 4chan. Using a one-month dataset covering 1.8M posts across seven communities, we quantified how users discuss policy, geopolitics, emerging AI technologies, and everyday topics, integrating Perspective API toxicity scores, sentiment polarity, readability metrics, and temporal activity patterns into a unified analysis. Our findings show clear cross-platform differences: 4chan exhibits significantly higher toxicity, shorter and more lexically varied posts, sharper spikes around policy events, and pronounced diurnal rhythms, while Reddit shows more coherent, longer, and less toxic discussions. Visualizations revealed model-specific AI discussion bursts, community-specific reactions to the U.S. tariff event, and highly structured posting patterns on `/pol/`, including a stable baseline, an extreme spike on 14 November, and predictable late-evening peaks. These measurements also surfaced a new research focus: engagement dynamics appear strongly linked to post type, linguistic style, and posting time, suggesting deeper structural factors behind interaction patterns.

Project 3 builds directly on these insights. We will move from descriptive measurement to explanatory analysis by examining how post formats, linguistic and stylistic features, and temporal factors shape engagement within these communities. In addition, we will develop an interactive tool that allows users to query the dataset and explore these effects dynamically, enabling reproducible, real-time analysis grounded in the metrics and visual patterns established in Project 2. :contentReference[oaicite:1]index=1

ACM Reference Format:
Karthik Maganahalli Prakash and Sanjana Shivanand. 2018. SocialMediaAvengers Project 3 Proposal. In . ACM, New York, NY, USA, 2 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Research Question

In Project 2, we moved from raw data collection to structured measurement across multiple Reddit and 4chan communities. We characterized temporal activity patterns, examined reactions to a specific U.S. tariff event, measured toxicity using the Google Perspective API, and analyzed sentiment and readability in AI-related discussions. Our results showed that 4chan and Reddit differ sharply in toxicity levels, text quality, and response to external events, and that within these platforms, activity often clusters around certain topics, keywords, and threads. We also observed that some posts and threads attracted disproportionately high engagement, suggesting that not all content is treated equally by the community.

These observations motivate a more focused question for Project 3: rather than only asking *when* people post or *how toxic* discussions are, we are interested in *what kind of posts* actually elicit responses and sustain conversations. Informally, our visualizations and exploratory plots from Project 2 indicated that posts that look like questions, strong opinions, or news links often generate more replies than short throwaway comments or low-effort content. However, this pattern was only observed qualitatively; we did not systematically test it.

To address this gap, Project 3 will concentrate on the following research question:

> **RQ1:** How do different types of posts (e.g., questions, opinions, news articles, memes) influence engagement and discussion within these communities?

We will operationalize this question by first categorizing posts into a small set of interpretable types based on their structure and intent, such as:

good! well thought out!

- **Question-style posts** (e.g., explicit questions, help-seeking, or prompts for discussion),
- **Opinion or commentary posts** (e.g., strong stances, reactions, or personal takes),
- **News or link-sharing posts** (e.g., URLs to articles, announcements, or reports),
- **Memes or low-text posts** (e.g., image-heavy or short, slang-heavy posts).

For each post type, we will measure engagement using concrete metrics such as reply count, thread length, and short-term reply velocity (e.g., number of responses within the first hour). We will also compare these patterns across platforms and communities (e.g., `/pol/` vs. r/geopolitics), leveraging the same data pipeline, time windows, and metadata defined in Project 2. By doing so, Project 3 moves beyond aggregate post counts and average toxicity levels and instead asks which formats actually succeed in capturing attention and sparking discussion. This research question directly builds on our ear

2 Analyses for the Interactive Tool

Our dashboard will make three Project 2 analyses interactive, with parameters directly tied to our dataset and RQ1:

(1) **/pol/ Activity Explorer:** Users can switch between daily and hourly activity on `/pol/` for the 1–14 November window, filter by post type, and compare how different formats behaved during the 14 November spike.

Conference'17, Washington, DC, USA
© 2018 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-XXXX-X/2018/06
https://doi.org/XXXXXXX.XXXXXXX

(2) **Policy-Event Reaction View:** The tariff-event analysis becomes interactive by letting users select the platform (4chan vs. Reddit), choose communities such as `/pol/`, `/int/`, or r/geopolitics, and adjust the event window (e.g., ±7 or ±30 days) to compare how each group reacted around the 9 November tariff announcement.

(3) **Toxicity–Readability–Engagement View:** Users can filter posts by Perspective API toxicity ranges, readability bands, and post-length bins to examine how these characteristics relate to reply counts and thread length across communities.

These analyses directly extend the temporal, toxicity, and event-based patterns identified in Project 2 and allow dynamic exploration of how post type and content features relate to engagement.

3 Tools, Libraries, and Frameworks

Project 3 builds directly on the infrastructure from Project 2 and adds a lightweight interactive layer for dynamic querying and visualization. The key tools are:

- **Python 3:** Used for all backend analysis tasks, including post-type labeling, engagement metric computation, and reproducing the temporal and toxicity analyses from Project 2.
- **PostgreSQL/TimescaleDB:** Stores all Reddit and 4chan posts, metadata, and Perspective API toxicity scores. TimescaleDB enables fast retrieval of hourly and daily windows for the interactive plots.
- **psycopg2:** Python's PostgreSQL adapter for issuing parameterized queries, supporting filters such as date range, platform, community, post type, and toxicity level.
- **pandas:** Handles merging toxicity/readability data, grouping posts by type, and computing engagement statistics for each analysis shown in the dashboard.
- **Flask:** Backend framework used to serve API endpoints that return filtered data in JSON format for the dashboard.
- **Bootstrap + HTML/CSS:** Provides the dashboard layout and UI controls for selecting communities, time windows, and content-based filters.
- **Plotly.js:** Generates interactive visualizations for the `/pol/` activity explorer, the tariff-event timeline, and the toxicity–engagement analysis.
- **matplotlib:** Used for generating static ACM-formatted figures in the Project 3 report.
- **systemd:** Keeps the Flask server running on the VM to ensure the dashboard is reliably accessible during evaluation.

This toolset allows us to reuse the core analyses from Project 2 while providing a responsive, interactive interface for exploring engagement patterns.

4 Implementation Plan

We will implement Project 3 in three layers: a Python analysis layer, a Flask API layer, and a React + Plotly.js dashboard.

Stage 1: Analysis Module (Python + pandas). We will write a reusable Python module that connects to PostgreSQL/TimescaleDB via `psycopg2`, assigns post-type labels (questions, opinions, news, memes), computes engagement metrics (reply count, thread depth, first-hour replies), and reproduces the key Project 2 outputs: `/pol/` daily/hourly activity, tariff-event timelines, and toxicity/readability summaries.

Stage 2: Backend API (Flask). We will expose this analysis logic through a small set of Flask endpoints (e.g., `/api/pol_activity`, `/api/tariff`, `/api/toxicity`) that accept parameters such as date range, community, post type, and toxicity band, then return JSON for the dashboard.

Stage 3: Front-End Dashboard and Deployment (React + Plotly.js). We will build a React-based dashboard with controls for selecting time windows, platforms/communities, post types, and toxicity/readability filters, and use Plotly.js to render the three interactive views. The Flask backend will be kept running on the VM via `systemd`, and the same analysis module will be used with `matplotlib` to generate static figures for the Project 3 report and the recorded demo.

5 Conclusion

Project 3 builds directly on the SocialMediaAvengers pipeline and the measurements from Project 2 by shifting from static plots to an interactive, question-driven analysis. Our goal is to answer RQ1—how different post types (questions, opinions, news, memes) influence engagement—using the same Reddit and 4chan data, toxicity scores, and temporal patterns already in place.

By turning the `/pol/` activity snapshots, tariff-event timeline, and toxicity/readability measurements into an interactive dashboard, we will allow users to explore how format, timing, and content features jointly shape replies, thread depth, and short-term activity. The implementation plan and tool choices outlined above ensure that the dashboard is tightly coupled to our existing database and analysis code, providing a clear and reproducible path from the Project 2 findings to a deeper understanding of engagement dynamics in Project 3.

References

[1] The pandas development team. 2020. *pandas: Python Data Analysis Library.* https://pandas.pydata.org

[2] John D. Hunter. 2007. Matplotlib: A 2D Graphics Environment. *Computing in Science & Engineering* 9, 3 (2007), 90–95.

[3] Armin Ronacher. 2010. *Flask: A Lightweight WSGI Web Application Framework.* https://flask.palletsprojects.com/

[4] Plotly Technologies Inc. 2015. *Plotly: Collaborative Data Science.* https://plot.ly

[5] Bootstrap Team. 2023. *Bootstrap: Front-End CSS Framework.* https://getbootstrap.com/

[6] Federico Di Gregorio. *psycopg2: PostgreSQL database adapter for Python.* https://www.psycopg.org/

[7] Timescale Inc. 2024. *TimescaleDB: An Open-Source Time-Series Database.* https://www.timescale.com/

[8] Google Jigsaw. 2023. *Perspective API.* https://perspectiveapi.com/

[9] Plotly. 2023. *plotly.js Documentation.* https://plotly.com/javascript/