

Censorship Detection

John Doros

john.doros10@bcmail.cuny.edu

Supervisor <Professor Chen, chen@sci.brooklyn.cuny.edu>

Abstract

Censorship detection is a ambitious challenge in understanding internet freedom as organizations use various forms of content restriction and blocking policies. This project aims to focus on preprocessing and data aggregation on censorship measurement data from Censored Planet to achieve a more accurate form of detecting censorship events. I will focus on processing raw censorship measurement data integrating metadata to it and applying aggregation methods to identify patterns in censorship behavior. My goal for this project is to develop an efficient data processing method and or approach that enables to identify potential censorship events while differentiating from false positives for example such as network outages. The understanding and methods developed from this project will contribute to a broader approach in efforts for censorship circumvention and or reliability in the future internet. This project will focus on supporting the CenDTect framework focusing on step 1 of CenDTect Architecture which is Raw data preprocessing, which transforms censorship measurement data into a more suitable structured format for a better approach to censorship analysis. Throughout this project the main goal is to advance the effectiveness of censorship detection methods and to contribute to a more reliable internet.

Research Focus & Hypothesis

This project focuses on how censorship measurement data can be processed and aggregated to improve the accuracy of detecting censorship to improve the accuracy of detecting censorship events while reducing the amount of false positives. By analyzing a large Censored Planet dataset I aim to develop a structured approach for classifying censorship events and be able to differentiate between actual censorship events and non censorship events such as network failures.

Research Question

How can censorship data be effectively aggregated and preprocessed to improve censorship detection accuracy while reducing false positives?

Hypothesis

If censorship data is preprocessed with aggregation techniques and with metadata integration then censorship events can become more accurately identified while reducing false positives.

Background on Censorship Mechanisms

Background on censorship mechanisms.

DNS Blocking: Preventing domain name lookups such as failing or redirecting

IP Blocking : Preventing connections to a IP address

HTTP/HTTPS Filtering: Blocking specific content on webpage

Network Disruptions: Packet injections or network outages

Previous Research in Censorship Detection

OONI : Volunteer based censorship measurement gathering organization

Censored Planet : Censorship measurement database using global vantage points

CenDTect : Decision tree based censorship event detection

Research Methodology

Data Collection

Extract censorship data from Censored Planted

Preprocessing and feature extraction

Cleaning the extracted data and extract metadata (ASN, HTTP response codes, ISP, TLS details)

Classification and testing

Categorize response as normal censorship, likely censorship or unlikely censorship, compare these detected events with OONI reports for validation

Aggregation and trend analysis

Grouping data by ISP, country or censorship type, and assess false positives (such as network outage)

Finally measure accuracy of censorship detection (by comparing with ground truth)

Tools

GitHub - Used for uploading all project related files, used for backup as well.

SQL - Used for storing and querying censorship data efficiently supporting data aggregation analysis.

Dropbox - Used for retrieving censorship data provided by my supervisor.

PySide2 (Python Library) - Used for creating simple interactive interfaces for visualizing censorship analysis if needed.

Matplotlib - Used for visualizing censorship analysis if needed.

OONI/Censys - Has real world censorship data trends used for cross checking my own data set to determine its accurateness.

Python - Programming language used for data processing/validation/aggregation makes it easier for data filtering

Data Sources

The data sources that will be used are as followed

Censored Planted “<https://censoredplanet.org/>” - Used for extracting raw censorship data measurement (Professor has provided a dropbox link where he had extracted this data so I can use it)

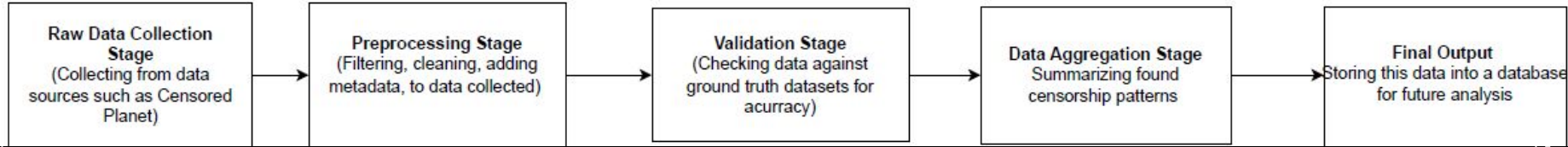
OONI “<https://ooni.org/>” - Possibly used for cross checking/validating censorship events for further accurateness

Citizen Lab “<https://citizenlab.ca/>” - Can be used for further checking/validating censorship events

ICANN “<https://www.icann.org/>” - Can be used for metadata extraction on IP address

WHOIS “<https://who.is/>” - Can be used for metadata extraction on IP address

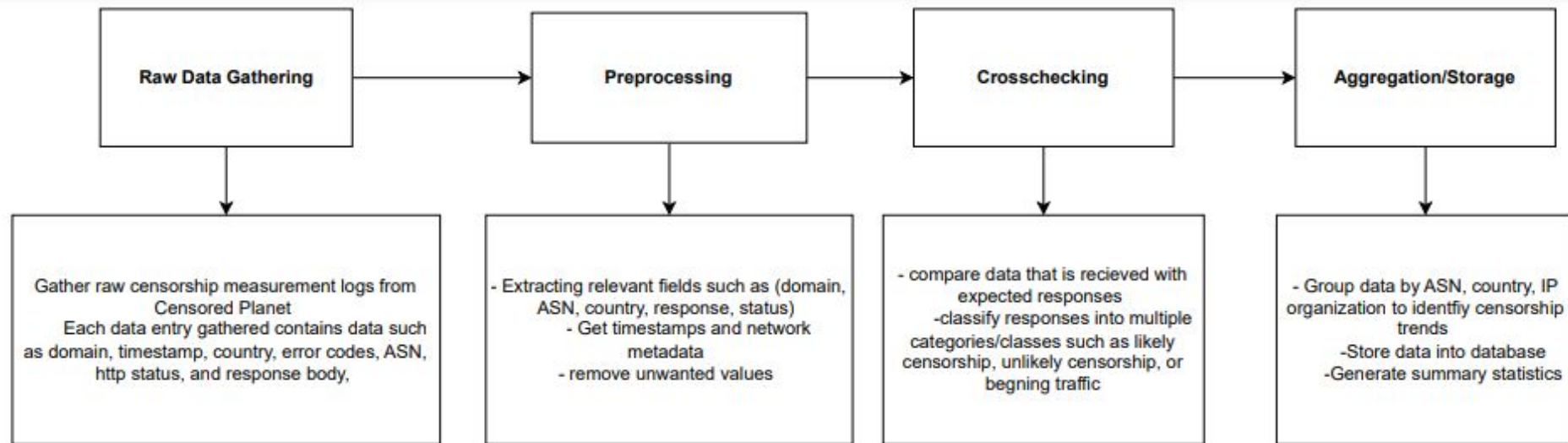
Stage Flow Diagram For Data Processing



Showing the data processing states at each stage and what will happen to the raw data at each stage.

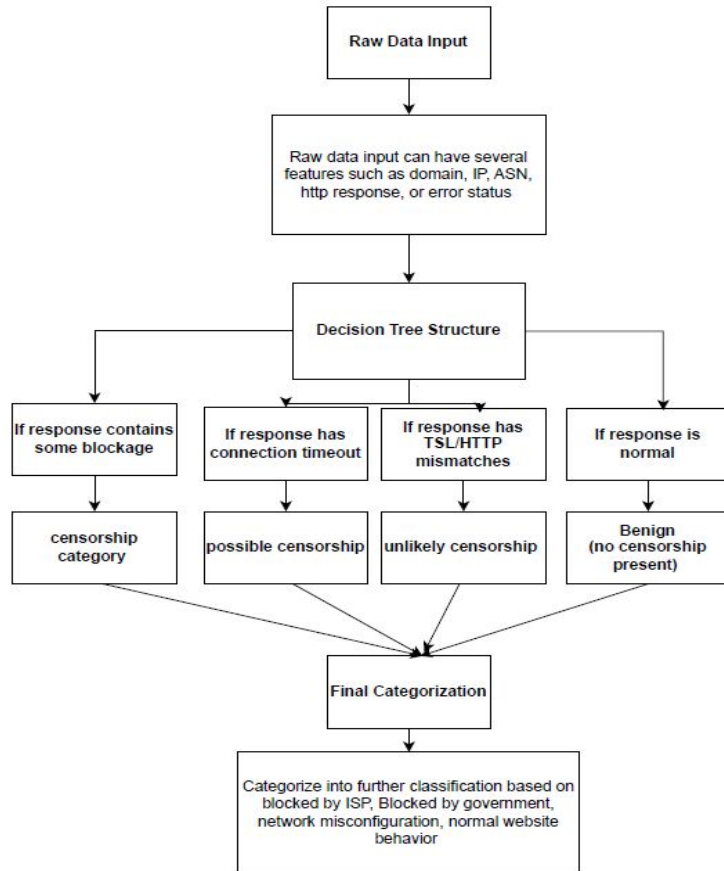
Logical flow of data processing (Sequence of operations on data)

Showing the what happens on processing pipeline in detail



Censorship event classification

How is a censorship event classified based on a decision tree flow, this would classify events based on different conditions, useful for demonstrating how events are analyzed to determine if censorship is likely or not into different classes



Use Cases

Some use cases for this technical implementation can be as follows

Identifying government website blocking : Governments in some areas block access to website contents to which can include news, social media and other websites to which governments do not like people to access at a particular time. This is important as people in society would form a bias towards the government and whether the government should be trusted or not.

Distinguishing between network outage and real censorship events : When it comes to detecting censorship events some events can create false positives to which the access to certain content are not caused by a censorship event but rather a network failure and or outage. This use case can minimize those false positives by distinguishing from a network error and a censorship event, this can have an impact on our data that we are processing to which can make our data more reliable and more accurate when it comes to detecting a censorship event.

Tracking censorship events over time : When it comes to a censorship event taking place such as during a political event, protests and or elections tracking censorship during these periods can be helpful for monitoring censorship trends over time to detect patterns and obtain further analysis. This can be useful to push for transparency during these periods and further can help detecting new censorship events and patterns.

February

Generally: Work on required documents presented on Blackboard, setup meetings with supervisor for project discussion

Feb 19. Reformat slides based on Supervisors meeting including a log file

Estimate time : 1 week

Feb 20.

Start updating on project management dashboard

Estimate time : 1 day

Feb 20-28. Start working on data filtering software

Estimate time until completion : 1 month

Organize version control repository and work on required Blackboard documents

Estimate time : 1 Week

First week of March:

Prepare and work on demos

Estimate time : 1 week

Continue to stay in touch with supervisor

Weekly basis

March:

Week 1 : Finish working on software

Estimate time : 1 week

Week 2 : Work on bug fixes in the software

Estimate time : 1 week

Work on deliverables required on Blackboard

Estimate time : 1 week

Rest of semester :

Continue to work on software, continue updating supervisor, continue to update project management tools

Estimate time : couple weeks

Tentative Schedule

Will continue to update this schedule but this is a rough idea
for now

Use Cases

Some use cases for this technical implementation can be as follows

Identifying government website blocking : Governments in some areas block access to website contents to which can include news, social media and other websites to which governments do not like people to access at a particular time. This is important as people in society would form a bias towards the government and whether the government should be trusted or not.

Distinguishing between network outage and real censorship events : When it comes to detecting censorship events some events can create false positives to which the access to certain content are not caused by a censorship event but rather a network failure and or outage. This use case can minimize those false positives by distinguishing from a network error and a censorship event, this can have an impact on our data that we are processing to which can make our data more reliable and more accurate when it comes to detecting a censorship event.

Tracking censorship events over time : When it comes to a censorship event taking place such as during a political event, protests and or elections tracking censorship during these periods can be helpful for monitoring censorship trends over time to detect patterns and obtain further analysis. This can be useful to push for transparency during these periods and further can help detecting new censorship events and patterns.

Expectations and Future Work

-Research expected contributions

Develop a structured data processing approach for detecting censorship

Created an improvement to reduce false positive

Provide trend analysis on censorship patterns over time

-Future considerations


Using machine learning models for censorship classification

Improve accuracy through combination of multiple datasets

Expand to focus on detecting new types of censorship

Github Repository

<https://github.com/john0x44/Censorship-Detection-In-Censored-Planet/tree/main>

 **Censorship-Detection-In-Censored-Planet** Public Pin Unwatch

main



















1 Branch

0 Tags

Go to file

Add file

Code

 john0x44	Add files via upload	12c2eb3 · 1 minute ago	11 Commits
 4900- time log week 1.pdf	Add files via upload	2 weeks ago	
 4900- time log week 2.pdf	Add files via upload	1 minute ago	
 4900_time_log_week_3.pdf	Add files via upload	1 minute ago	
 CHANGELOG	Create CHANGELOG	last week	
 CNDETECT_2024-409-paper.pdf	Add files via upload	2 weeks ago	
 CamScanner 02-10-2025 20.43_1.jpg	Add files via upload	2 weeks ago	
 CamScanner 02-10-2025 20.43_2.jpg	Add files via upload	2 weeks ago	
 CamScanner 02-10-2025 20.43_3.jpg	Add files via upload	2 weeks ago	
 CamScanner 02-10-2025 20.43_4.jpg	Add files via upload	2 weeks ago	
 CamScanner 02-10-2025 20.43_5.jpg	Add files via upload	2 weeks ago	
 CamScanner 02-10-2025 20.43_6.jpg	Add files via upload	2 weeks ago	
 DIAGRAM_1.pdf	Add files via upload	last week	
 Diagram_2.pdf	Add files via upload	last week	
 Diagram_3.pdf	Add files via upload	last week	
 Notes_1.pdf	Add files via upload	2 weeks ago	
 README.md	Initial commit	2 weeks ago	
 meeting_1_notes_2_11.jpg	Add files via upload	last week	

Project Management

<https://github.com/users/john0x44/projects/1/views/1>

Censorship Detection

Backlog

Priority board

Team items

Roadmap

In review

My items

New view

Filter by keyword or by field

Discard

Save

Backlog

9 / 20

Estimate: 0

...

This item hasn't been started

processing

Censorship-Detection-In-Censored-Planet #9

Define a validation process for Censorship Events

Censorship-Detection-In-Censored-Planet #10

Implement a HTTP response categorization strategy

Censorship-Detection-In-Censored-Planet #11

Developing an aggregation process strategy

Censorship-Detection-In-Censored-Planet #12

Storing into a database

Censorship-Detection-In-Censored-Planet #13

Creating visual reports for processed data

Censorship-Detection-In-Censored-Planet #16

reorganize github repository files

+ Add item

Ready

0

Estimate: 0

...

This is ready to be picked up

+ Add item

In progress

5 / 20

Estimate: 0

...

This is actively being worked on

Censorship-Detection-In-Censored-Planet #1

Reformat slides to a more scientific research approach

Censorship-Detection-In-Censored-Planet #2

Finish Project Tools Survey

Censorship-Detection-In-Censored-Planet #3

Updating kanban dashboard

Censorship-Detection-In-Censored-Planet #14

Processing raw censorship data

Censorship-Detection-In-Censored-Planet #15

Implementing minibatch data processing

+ Add item

In review

0 / 10

Estimate: 0

...

This item is in review

+ Add item

Done

2

Estimate: 0

...

This has been completed

Censorship-Detection-In-Censored-Planet #5

Had setup github repository

Censorship-Detection-In-Censored-Planet #4

Timeelog for 2/24

+ Add item