1-800-Censorship: Analyzing internet censorship data using the Internet Yellow Pages

Friedemann Lipphardt
Max Planck Institute for Informatics
Internet Architecture
Saarbruecken, Germany
frlippha@mpi-inf.mpg.de

Malte Tashiro
IIJ Research Laboratory / SOKENDAI
Tokyo, Japan
malte@iij.ad.jp

Romain Fontugne
IIJ Research Laboratory
Tokyo, Japan
romain@iij.ad.jp

Abstract

The Internet Yellow Pages (IYP) is a knowledge graph unifying access to various Internet datasets via a common query language, Cypher. Hence, interested researchers can easily conduct analysis across numerous datasets without having to implement tools to parse and handle raw data from various sources. In order to add Internet censorship data to IYP, this paper presents our efforts to integrate the Open Observatory of Network Interference (OONI) data to IYP. It also demonstrates how one can effectively utilize the integrated data to jointly analyze OONI data with other datasets made available by IYP.

Keywords

Internet censorship, knowledge graph, Internet routing

1 Introduction

Internet censorship involves complex systems deployed by some countries to control and restrict the flow of information to their citizens. State authorities and regulatory bodies deploy a range of techniques, including DNS tampering, IP filtering, deep packet inspection, and the use of sophisticated middleboxes that inspect and selectively block traffic.

The detection and research of these techniques often involve a mix of remote measurements and on-the-ground testing. A popular measurement project facilitating these tests is the Open Observatory of Network Interference (OONI) [1], which enables volunteers to perform tests themselves on a variety of devices by providing easy-to-use applications. As a consequence, OONI produces large amounts of data, over 36 million measurements in March 2025, making the analysis challenging.

To simplify analysis and integration with other datasets, we integrate OONI data into the Internet Yellow Pages (IYP) [5]. IYP is a knowledge graph that unifies various Internet datasets and makes them accessible via a specialized query language. By combining the crowd-sourced censorship data from OONI with the expansive collection of datasets contained in IYP, we enable interested parties to explore new avenues of censorship research without the need for repeated manual data integration.

Free and Open Communications on the Internet 2025(2), 10–17

2 Background

2.1 OONI

The Open Observatory of Network Interference (OONI) is a widely deployed platform that monitors Internet censorship through a series of distributed measurement probes. OONI's tests examine various layers of the network stack, from DNS resolution anomalies to HTTP(S) connectivity disruptions. Each test is described by a measurement specification, which is documented on GitHub [17], and results in a single dataset.

Although OONI's data is publicly available on their website [16] and via an API [15], it suffers the same shortcomings as other datasets, requiring interested researchers to either fully rely on the provided data presentations, or implement their own scripts to process the data given by the API, and expend a lot of work to link it to other datasets.

2.2 Internet Yellow Pages & Cypher

The Internet Yellow Pages is a knowledge graph for Internet resources. Currently, it combines 48 datasets from 24 organizations into a single knowledge graph. The datasets span a variety of sources, from autonomous system (AS) information over IP allocations to DNS resolution data. Integration into a unified knowledge graph enables homogeneous access to the data.

Since IYP is based on Neo4j, familiarity with the Cypher query language is required to access the data. A full description of Cypher is outside the scope of this article, but we describe the basic components of a query here. Queries usually consist of keywords, nodes `()`, and relationships `[]`.

Important keywords are `MATCH`, `WHERE`, `WITH`, and `RETURN`. `MATCH` is similar to `SELECT` in SQL, it is followed by a search pattern (explained below) and describes the data that should be retrieved. `MATCH` can be used multiple times to query for different patterns and only data that matches all patterns is retrieved. A `MATCH` can also be marked as `OPTIONAL` in which case the pattern does note have to be matched. `WHERE` is used to apply filters based on nodes or relationship properties. `WITH` enables intermediate aggregation of data for further processing

and is semantically similar to `RETURN`, which finishes the query and specifies which parts of the data is returned.

Nodes and relationships have a type that can be specified after a colon, e.g., `(:HostName)` or `[:RESOLVES_TO]`. Optionally, they can be bound to a variable name that is placed before the colon `(h:HostName)`, which in turn gives access to properties of the node/relationship. Assigning a variable name to a node also allows to reuse the node in other parts of the query. One major feature of Cypher is the expression of queries in form of search patterns. A pattern is a visual representation of the subgraph that should be searched for by the query. For example, the pattern

```
(:HostName)-[:RESOLVES_TO]->(:IP)
```

would search for host names in the graph and the corresponding resolving IP addresses. Patterns can be chained together as a sequence of nodes connected by relationships.

2.3 Related Work

A recent paper by Wendzel et al. [21] surveys the global state and techniques of Internet censorship, as well as available datasets. It provides a good overview of the potential research topics that interested parties can explore using IYP. It also gives insights into relevant datasets, some of which are already present in IYP, including OONI, and some that are not.

Fletcher et al. [4] examine the tradeoff between expert analysis and remote measurements in Internet censorship data, giving insight into why the remote measurements we get from OONI are not enough to make judgments, and expert analysis such as utilizing IYP, are a useful addition.

Diving into measurement techniques, Quack [20] by VanderSloot is now one of the most commonly used techniques to detect HTTP and TLS based censorship.

Citizenlab [11] provides a comprehensive testlist to probe for website censorship which is used as part of the OONI test suite [14].

Lee et al. [12] examine the impact of Tor on censorship circumvention, another tool that is tested using the OONI test suite.

3 OONI Integration into IYP

The integration of datasets into IYP is done via crawlers [6, 8]. A crawler is a script that fetches the data, transforms it according to IYP's ontology, and inserts the transformed data into the knowledge graph. In order to add OONI data to IYP we have to revise the ontology to include censorship data, and for each dataset create a graph representation and implement the corresponding crawler.

We implemented a dedicated IYP crawler for each OONI dataset. While each crawler is designed specifically around the corresponding OONI specification, they share the same logic for processing and aggregating data. In total, we integrated 13 tests, only excluding performance related tests (DASH & NDT). In accordance with the weekly IYP snapshots, all crawlers fetch one week of data.

We import data exactly as crawled from OONI, without any filtering. Because censorship tests may be ambiguous and open to interpretation, many OONI tests results are reported as Anomaly, Failure, or OK, without explicitly saying if a content is censored or not. Our crawlers replicate this categorization so that different interpretations of the data is still possible with IYP. One particularly important test for the rest of this paper is the Webconnectivity test, detailed in [17] under the /nettests/ts-017-web-connectivity.md subfolder. This test queries a subset of URLs sourced from CitizenLab [14] for blockpages, with results classified into four categories: OK (no blockpage detected), Anomaly, Failure, and Confirmed (indicating confirmed censorship, a blockpage encountered). An OK result indicates the fetched URL matches what has been fetched by a control server. This test and its classification are significant in Section 4. We now briefly describe how the crawlers are implemented.

In the graph representation all tests are modeled as a `CENSORED` relationship connecting an `AS` node to a target. The AS node represents the source of the measurements, i.e., the network in which OONI probes are located. The type of the target node depends on the test. We aggregate the results obtained for a week of data by AS and target, and attach the aggregated results to the CENSORED relationship. The exact result categories depend on the test, but for each category we provide an aggregated count and a percentage to simplify queries.

The `webconnectivity` and `STUN reachability` tests target `URL` nodes. As previously mentioned the webconnectivity test provides four result categories, whereas STUN gives only two categories (succeeded, failed). For `Tor` the targets are IP addresses, so we create a relationship to `IP` nodes. Additionally, we connect the IP with a `Tag` node using a `CATEGORIZED` relationship where the tag indicates which kind of Tor directory or bridge is operated at the IP. Tests that lack an explicit target list, such as the `Psiphon` or `RiseupVPN` tests, are implemented differently. For these, we introduce a single central `Tag` node named after each test (e.g., Psiphon, RiseupVPN). This approach of one central Tag node linked to multiple ASes, is the most frequently employed pattern among the crawler implementations.

For better integration with other IYP datasets, we also include data that is not directly related to censorship, such as DNS resolution for hostnames (webconnectivity, STUN), the IPs of DNS resolvers (webconnectivity) and the country in which an AS is present (based on the probe location).

More information on the specifics of our implementation can be found on the IYP GitHub repository [10].

4 Results

This paper mainly serves as a report for the addition of internet censorship data in IYP and a guideline for further expansion and future work, an introduction into best practices for such a task, as well as an inspiration for the possible angles to examine censorship using IYP. By doing so, we enable researchers to easily compare datasets, enabling connections and queries across datasets without doing the manual work of setting them

all up themselves. See Listing 2 for an example of such an advanced query. We also present some preliminary results, including the queries we used to gather these results, intending to serve as an inspiration to like-minded researchers aiming to better understand Internet censorship. However, since this is primarily a tool-based paper, not a censorship work, we do not aim to fully detailed or explain the observed phenomena. We encourage interested parties to use the provided queries as a baseline for further research. All of these queries can be tried on the live instance of IYP [9]. Also as IYP is weekly updated, the queries can be used to obtain up-to-date results.

4.1 Censorship Leakage

Motivation: We define the concept of censorship leakage to be the unintended presence of censorship, i.e. DNS traffic transiting through a country and being affected by that countries DNS censorship [18]. To find this, we query IYP to retrieve countries where the censorship rate are known to be low or nonexistent and that have border countries that are known to have a high censorship rate. This query can be further refined by looking at specific types of censorship, which is easily done by examining the link type, i.e., DNS-based censorship, HTTP-based censorship or similar. This serves as a generalized example to build upon. In this example we look at neighboring countries of Russia. While the presence of censorship in China isn't unexpected, we specifically added it, as well as Russia itself, to the list as a comparative example. North Korea is excluded for obvious censorship presence and lack of data.

Query: See Listing 1 in the Appendix.

Result: Table 1 shows the result of the query as of February 2025, neighboring countries to Russia and their censorship rate, defined as (100% - the aggregated average percentage of unblocked queries). Countries not listed didn't have data during the most recent IYP run.

Country	Censorship Rate (%)	Total Test Count
China	77.66	246 279
Russian Federation	43.59	778 410
Lithuania	21.73	9157
Belarus	19.15	9160
Ukraine	14.87	110 144
Kazakhstan	14.26	41 700
Norway	12.04	142 019
Finland	12.03	105 537
Azerbaijan	11.66	8137
Poland	9.74	244 024
Georgia	9.43	97 183
Estonia	9.39	31 179
Latvia	7.24	11 565

Table 1: Censorship Rates and Total Test Counts for Selected Countries

4.2 Transit Censorship

Motivation: Another recent finding discovered the presence of Russian Internet censorship on packets that were simply transiting through the countries' infrastructure [2]. We select the ASNs named in [2] to find more potential cases of transit censorship by using a more advanced query, only made possible through the IYP's sophisticated query language and implementation of a combination of various datasets. We select all of the URLs that were intercepted via a blockpage when accessed from the aforementioned ASNs with a blockage rate of at least 90%. Then, we leverage other datasets in IYP and only consider popular URLs that are in the top 10k of ranking lists like Cisco Umbrella [3] or Google's Chrome User Experience Report [7]. Then, we query all ASNs that these ASNs peer with which are not located in Russia, China or North Korea and evaluate their blocking rate for these URLs. We take it one step further by filtering all other ASNs in the country of the peering ASN for blockage rate of 90% of that URL, to eliminate URLs that are blocked in that country for other reasons.

Query: See Listing 2 in the Appendix.

Result: Table 2 shows the result of the query as of February 2025, a selection of ASNs peering with the ASNs confirmed to be employing transit censorship, sorted by highest percentage of confirmed blockpages shown (see column Avg. Confirmed %, which is the average percentage of confirmed blockpages across all tests done in that ASN). As explained above, these URLs are not blocked in other ASNs in these countries. We also found Listing 5 interesting, the most common URLs found to be transit censored. Due to space constraints, these URLS and the query used to produce this result can be found in the Appendix A.

ASN	Country	# URLs	Avg. Confirmed (%)
3214	Germany	4	75
6939	Sweden	4	75
216071	Netherlands	38	59
9123	Netherlands	162	57
49127	Netherlands	30	53
9605	Japan	4	50
18001	Sri Lanka	9	33
34549	Malaysia	74	24
1273	United Kingdom	5	20
9121	Türkiye	74	20
8452	Egypt	97	16
211597	United Kingdom	44	16
3356	United States of America	11	11
45899	Viet Nam	74	10
15802	United Arab Emirates	10	10
13285	United Kingdom	74	8
17557	Pakistan	74	8
50266	Netherlands	74	8
1267	Italy	74	8
8781	Qatar	52	8

Table 2: ASNs Suffering Suspected Transit Censorship

4.3 Unexpected Censorship

Motivation: By taking the average per country of all types of censorship found in the dataset, we can find a list of countries which are not typical culprits of Internet censorship. In the query below, we exclude the most common censored countries (China, Iran, North Korea, Russia, Myanmar, Iraq, India, Pakistan, Egypt) and list all countries with a unblocked average of less than 75%, therefore blocking more than 25% of all probes.

Query: See Listing 3 in the Appendix.

Result: Table 3 shows the result of the query as of February 2025, the countries with the highest rate of blocked queries across all censored link types, therefore not distinguishing between potential censorship types.

Country	Avg. Unblocked (%)	Total Test Count
Venezuela	68.83	461114
Cuba	69.70	5501

Table 3: Average Unblocked Rates for Selected Countries

4.4 High Failure Rates

Motivation: Another interesting query is looking at URLs within countries that are known to censor, such as China, but are not confirmed to be censored via blockpage but still have a high percentage of either anomaly or failure, therefore indicating anomalous behavior. To do that, we can filter for URLs withing the country we are interested in, in this case China, and specifically filter for ones with high failure or anomaly and low confirmed and OK rate.

Query: See Listing 4 in the Appendix.

Result: Table 4 shows the result of the query as of February 2025. Due to space constraints we exclude most of the columns in the query above, only including asn_count, total_failure, and average_failure as most other entries were 0. Avg. Failure % is the average failure rate across all tests done from those countries, 100% meaning every single test for that URL from that country failed.

URL	ASN Count	Total Failure Count	Avg. Failure (%)
https://hkleaks.ru/	3	8	100
https://blockdx.co/	4	13	100
http://www.tobacco.org/	3	8	100
https://www.yuemei.com/	4	10	100
https://www.igengmei.com/	3	8	100
http://www.dit-inc.us/	7	21	100
http://www.fordfound.org/	7	20	100
http://vho.org/	3	8	100
https://www.humanflow.com/	4	10	100
https://libgen.space/	4	9	100

Table 4: Summary of Failures Sorted by Failure Rate, Top 10

5 Conclusion & Future Work

5.1 Conclusion

In this paper we introduced our implementation of the OONI dataset into IYP, allowing interested censorship researchers to make easy use of the powerful tools and datasets provided by the combination. Further, we show how to use the OONI dataset and the IYP Cypher language in general. We also introduce a few initial findings, as well as their Cypher queries, establishing a basis for future work. By adding censorship data to the IYP, we enable access advanced cross-dataset queries such as Listing 2 or Listing 1. to any interested researchers, without the need for them to implement their own crawlers, API pipeline or comparison engine, the IYP handles all of that for them, and packages it an easily accessible Cypher language.

5.2 Future Work

Complementary to OONI, several other measurement platforms have emerged to enhance our understanding of network interference. Platforms such as ICLab [13], or Censored Planet [19] have developed methodologies to track censorship trends via remote vantage points and longitudinal analyses.

These alternatives often focus on examining high-level indicators, such as domain reachability and server response behavior, but may overlook granular, application-specific filtering details. Furthermore, these applications do not employ the volunteer-based approach we see used by OONI, instead relying on different techniques to detect censorship using specific vantage points. These sites are therefore angled more towards expanding the capabilities of OONI in giving censorship insight, rather than fully replacing it.

An implementation of these services into the IYP would further expand the capabilities of researchers to analyze existing censorship data, by enabling easy cross-comparison across the ICLab, Censored Planet and OONI datasets, as well as non-censorship datasets already available in the IYP, without having to implement dataset crawlers themselves. The authors have reached out to the owners of these projects to facilitate the use of their data to this end, but at time of writing have unfortunately received no response. Therefore, we leave this aspect to future work.

Acknowledgments

The authors are grateful to the anonymous reviewers for their constructive comments.

References

[1] 2012. OONI: Open Observatory of Network Interference. In *2nd USENIX Workshop on Free and Open Communications on the Internet (FOCI 12)*. USENIX Association, Bellevue, WA. https://www.usenix.org/conference/foci12/workshop-program/presentation/Filast{ò}

[2] Dave Levin Aaron Ortwein, Kevin Bock. 2023. Towards a Comprehensive Understanding of Russian Transit Censorship. In *Free and Open Communications on the Internet Workshop (FOCI)*. https://www.cs.umd.edu/~dml/papers/transit_censorship_foci23.pdf Accessed: 2025-04-08.

[3] Cisco. 2025. Umbrella Popularity List. https://s3-us-west-1.amazonaws.com/umbrella-static/index.html Accessed: 2025-04-08.

[4] Terry Fletcher and Andria Hayes-Birchler. 2020. Comparing Measures of Internet Censorship: Analyzing the Tradeoffs between Expert Analysis and Remote Measurement. https://doi.org/10.5281/zenodo.3967398
[5] Romain Fontugne, Malte Tashiro, Raffaele Sommese, Mattijs Jonker, Zachary S. Bischof, and Emile Aben. 2024. The Wisdom of the Measurement Crowd: Building the Internet Yellow Pages a Knowledge Graph for the Internet. In *Proceedings of the 2024 ACM on Internet Measurement Conference* (Madrid, Spain) *(IMC '24)*. Association for Computing Machinery, New York, NY, USA, 183–198. https://doi.org/10.1145/3646547.3688444
[6] Romain Fontugne, Malte Tashiro, Raffaele Sommese, Mattijs Jonker, Zachary S. Bischof, and Emile Aben. 2025. Internet Yellow Pages (IYP) Tutorial. https://docs.google.com/document/d/1PdOEkaep2wBkTR2ZVqN_f7MdDll3shki88S-TKjz074/ Accessed: 2025-04-08.
[7] Google. 2025. Chrome User Experience Report. https://developer.chrome.com/docs/crux Accessed: 2025-04-08.
[8] Internet Health Report Project. 2025. Internet Yellow Pages. https://github.com/InternetHealthReport/internet-yellow-pages Accessed: 2025-04-08.
[9] Internet Initiative Japan Inc. (IIJ). 2025. Internet Yellow Pages (IYP). https://iyp.iijlab.net/ Accessed: 2025-04-08.
[10] Internet Yellow Pages Project. 2025. OONI Crawler — Internet Yellow Pages. https://github.com/InternetHealthReport/internet-yellow-pages/tree/main/iyp/crawlers/ooni Accessed: 2025-06-17.
[11] Citizen Lab and Others. 2014. URL testing lists intended for discovering website censorship. https://github.com/citizenlab/test-lists https://github.com/citizenlab/test-lists.
[12] Linda Lee, David Fifield, Nathan Malkin, Ganesh Iyer, Serge Egelman, and David Wagner. 2016. *Tor's Usability for Censorship Circumvention*. Master's thesis. EECS Department, University of California, Berkeley. http://www2.eecs.berkeley.edu/Pubs/TechRpts/2016/EECS-2016-58.html
[13] Arian Akhavan Niaki, Shinyoung Cho, Zachary Weinberg, Nguyen Phong Hoang, Abbas Razaghpanah, Nicolas Christin, and Phillipa Gill. 2020. ICLab: A Global, Longitudinal Internet Censorship Measurement Platform. In *2020 IEEE Symposium on Security and Privacy (SP)*. 135–151. https://doi.org/10.1109/SP40000.2020.00014
[14] Open Observatory of Network Interference (OONI). 2025. Citizen Lab - OONI Partnership. https://ooni.org/partners/citizen-lab/ Accessed: 2025-04-08.
[15] Open Observatory of Network Interference (OONI). 2025. OONI API. https://api.ooni.io/ Accessed: 2025-04-08.
[16] Open Observatory of Network Interference (OONI). 2025. OONI Explorer. https://explorer.ooni.org/ Accessed: 2025-04-08.
[17] Open Observatory of Network Interference (OONI). 2025. OONI Measurement Specifications. https://github.com/ooni/spec/tree/master Accessed: 2025-04-08.
[18] Neo Sparks, Smith Tank, and Dozer. 2012. The Collateral Damage of Internet Censorship by DNS Injection. *SIGCOMM Computer Communication Review* 42, 3 (2012), 21–27. https://conferences.sigcomm.org/sigcomm/2012/paper/ccr-paper266.pdf Accessed: 2025-04-08.
[19] Ram Sundara Raman, Prerana Shenoy, Katharina Kohls, and Roya Ensafi. 2020. Censored Planet: An Internet-wide, Longitudinal Censorship Observatory. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security* (Virtual Event, USA) *(CCS '20)*. Association for Computing Machinery, New York, NY, USA, 49–66. https://doi.org/10.1145/3372297.3417883
[20] Benjamin VanderSloot, Allison McDonald, Will Scott, J. Alex Halderman, and Roya Ensafi. 2018. Quack: Scalable Remote Measurement of Application-Layer Censorship. In *27th USENIX Security Symposium (USENIX Security 18)*. USENIX Association, Baltimore, MD, 187–202. https://www.usenix.org/conference/usenixsecurity18/presentation/vandersloot
[21] Steffen Wendzel, Simon Volpert, Sebastian Zillien, Julia Lenz, Philip Rünz, and Luca Caviglione. 2025. A Survey of Internet Censorship and its Measurement: Methodology, Trends, and Challenges. arXiv:2502.14945 [cs.CR] https://arxiv.org/abs/2502.14945

A Extra Queries

In this appendix we provide the Cypher queries for the results highlighted in the paper.

```
MATCH (c:Country)
      -[:COUNTRY {reference_org: 'OONI'}]-
      (:AS)-[x:CENSORED]->(target)
WHERE c.name IN ['Norway','Finland','Estonia',
    'Latvia','Lithuania','Poland','Belarus','
    Ukraine','Georgia','Azerbaijan','
    Kazakhstan','Mongolia', 'China', 'Russian
    Federation']
WITH c, x,
     COALESCE(
       // Facebook Messenger
       x.percentage_unblocked,
// Header Field Manipulation (no_total)
       x.percentage_no_total,
// HTTP Invalid Requestline (no_tampering)
       x.percentage_no_tampering,
// Signal, RiseupVPN, STUN, Tor, TORSF,
    Vanilla TOR, Web Connectivity
       x.percentage_ok,
       // Psiphon (working)
       x.percentage_working,
       // Telegram (total_ok)
       x.percentage_total_ok,
       // Telegram & WhatsApp (web_ok)
       x.percentage_web_ok,
       // Telegram (http_ok)
       x.percentage_http_ok,
       // Telegram (tcp_ok)
       x.percentage_tcp_ok,
       // WhatsApp (endpoint_ok)
       x.percentage_endpoint_ok,
       // WhatsApp (registration_server_ok)
       x.percentage_registration_server_ok
     ) AS testUnblocked
WITH c,
     sum(x.total_count * testUnblocked) AS
         weightedUnblocked,
     sum(x.total_count) AS totalTestCount
WITH c, (weightedUnblocked / totalTestCount)
    AS avgUnblocked, totalTestCount
RETURN c.name AS countryName,
       (100.0 - avgUnblocked) AS
           censorship_rate,
       totalTestCount AS total_test_count
ORDER BY censorship_rate DESC
```

Listing 1: Censorship Leakage

```
// Get top URLs censored from ASes
MATCH (base:AS)-[r1:CENSORED]->(url:URL)
      -[:PART_OF]->(:HostName)
      -[ra:RANK]->(:Ranking)
WHERE base.asn IN [3216, 25227, 35816, 47203,
    60299, 201776]
  AND r1.percentage_confirmed >= 90
  AND ra.rank <= 10000
WITH DISTINCT base, url
// Get peer ASes that also observe censorship
    and are not from Russia or China
MATCH (base)-[:PEERS_WITH]-(peer:AS)
      -[r2:CENSORED]->(url)
MATCH (peer_c:Country)
      -[:COUNTRY {reference_org:'OONI'}]-
      (peer)
WHERE NOT peer_c.country_code IN ['RU','CN']
WITH DISTINCT url, peer, peer_c, r2
// Ignore  URLs that are blocked in the entire
     country.
OPTIONAL MATCH (peer_c)-[:COUNTRY]-(other:AS)
    -[rOther:CENSORED]->(url)
WHERE other.asn <> peer.asn
  AND NOT other.asn IN [3216, 25227, 35816,
      47203, 60299, 201776]
WITH url, peer, peer_c, r2, collect(rOther.
    percentage_confirmed) AS otherRates
WHERE NOT (size(otherRates) > 0
  AND all(x IN otherRates WHERE x >= 90))
WITH peer, peer_c,
  count(DISTINCT url) AS numUrls,
  avg(r2.percentage_confirmed) AS avgConfirmed
RETURN peer.asn AS asn,
  peer_c.name AS country,
  numUrls, avgConfirmed
ORDER BY avgConfirmed DESC
```

Listing 2: Transit Censorship

```
MATCH (c:Country)
      -[:COUNTRY {reference_org: 'OONI'}]-
      (:AS)-[x:CENSORED]->(target)
WHERE NOT c.name IN [
  "China",
  "Iran, Islamic Republic of",
  "North Korea",
  "Russian Federation",
  "Myanmar",
  "Iraq",
  "India",
  "Pakistan",
  "Egypt"
]
WITH c, x,
  COALESCE(
    // e.g. Facebook Messenger
    x.percentage_unblocked,
    // Header Field Manipulation (no_total)
    x.percentage_no_total,
    // HTTP Invalid Requestline (no_tampering)
    x.percentage_no_tampering,
    // Signal, RiseupVPN, STUN, Tor, TORSF,
        Vanilla TOR, Web Connectivity
    x.percentage_ok,
    // Psiphon (working)
    x.percentage_working,
    // Telegram (total_ok)
    x.percentage_total_ok,
    // Telegram & WhatsApp (web_ok)
    x.percentage_web_ok,
    // Telegram (http_ok)
    x.percentage_http_ok,
    // Telegram (tcp_ok)
    x.percentage_tcp_ok,
    // WhatsApp (endpoint_ok)
    x.percentage_endpoint_ok,
    // WhatsApp (registration_server_ok)
    x.percentage_registration_server_ok
  ) AS testUnblocked
WITH c,
  sum(x.total_count * testUnblocked) AS
      weightedUnblocked,
  sum(x.total_count) AS totalTestCount
WITH c, weightedUnblocked / totalTestCount AS
    avgUnblocked, totalTestCount
WHERE avgUnblocked < 75
RETURN c.name AS countryName, avgUnblocked,
    totalTestCount
ORDER BY avgUnblocked ASC
```

Listing 3: Unexpected Censorship

```
MATCH (a:AS)-[r:CENSORED]->(url)
MATCH (a)
      -[:COUNTRY {reference_org: 'OONI'}]->
      (c:Country)
WHERE c.name = 'China'
WITH url,
     avg(r.percentage_anomaly) AS avg_anomaly,
     avg(r.percentage_failure) AS avg_failure,
     avg(r.percentage_confirmed) AS
         avg_confirmed,
     avg(r.percentage_ok) AS avg_ok,
     sum(r.count_anomaly) AS total_anomaly,
     sum(r.count_failure) AS total_failure,
     sum(r.count_confirmed) AS
         total_confirmed,
     sum(r.count_ok) AS total_ok,
     count(DISTINCT a) AS asn_count
WHERE avg_anomaly >= 90 OR avg_failure >= 90
  AND avg_confirmed <= 10 AND avg_ok <= 10
RETURN url.url AS url,
       asn_count,
       total_anomaly,
       total_failure,
       total_confirmed,
       total_ok,
       avg_anomaly,
       avg_failure,
       avg_confirmed,
       avg_ok
ORDER BY avg_anomaly DESC, avg_failure DESC
```

Listing 4: High Failure Rates

```
MATCH (base:AS)-[r1:CENSORED]->(url:URL)
      -[:PART_OF]->(:HostName)-[ra:RANK]->
      (:Ranking)
WHERE base.asn IN
    [3216,25227,35816,47203,60299,201776]
  AND r1.percentage_confirmed >= 90
  AND ra.rank <= 10000
WITH DISTINCT url
MATCH (p:AS)-[r2:CENSORED]->(url)
WHERE NOT p.asn IN
    [3216,25227,35816,47203,60299,201776]
MATCH (peer_c:Country)
  -[:COUNTRY {reference_org:'OONI'}]-(p)
WHERE NOT peer_c.country_code IN ['RU','CN', '
    KP']
WITH url, p, peer_c, r2
OPTIONAL MATCH (peer_c)-[:COUNTRY]-(other:AS)
    -[rOther:CENSORED]->(url)
WHERE other.asn <> p.asn
  AND NOT other.asn IN
      [3216,25227,35816,47203,60299,201776]
WITH url, p, peer_c, r2, collect(rOther.
    percentage_confirmed) AS otherRates
WHERE NOT (size(otherRates) > 0
  AND all(x IN otherRates WHERE x >= 90))
WITH url,
  avg(r2.percentage_confirmed) AS avgConfirmed
ORDER BY avgConfirmed DESC
WITH collect(
  {url: url,
   avgConfirmed: avgConfirmed}
  ) AS urlList,
  max(avgConfirmed) AS maxConfirmed
UNWIND urlList AS entry
WITH entry, maxConfirmed
WHERE entry.avgConfirmed = maxConfirmed
RETURN entry.url.url AS url, entry.
    avgConfirmed AS avgConfirmed
```

Listing 5: Transit Censorship URLs

URL
https://informator.ua/uk
https://politis.com.cy/
https://orf.at/
https://www.hrw.org/video-photos/interactive/2013/02/01/people-and-power-2014-sochi-olympics/
https://www.ionos.com/
https://www.hrw.org/report/2013/02/06/race-bottom/exploitation-migrant-workers-ahead-russias-2014-winter-olympic-games/
https://www.rainews.it/
https://twitter.com/GraniTweet/
https://twitter.com/openrussia_org/
https://twitter.com/15_minut/
https://observador.pt/
https://twitter.com/ATR__Official/
https://turbovpn.com/
https://www.freecity.lv/
https://www.facebook.com/navalny/
https://www.facebook.com/atrchannel/
https://www.facebook.com/GraniRu/
https://nv.ua/
https://newtime.ua/
https://newsmaker.md/
https://antikor.com.ua/
https://www.golosameriki.com/
https://lb.ua/
https://adguard.com/

Table 5: List of commonly censored popular URLs