Poster: Revealing Hidden Secrets: Decoding DNS PTR records with Large Language Models

Kedar Thiagarajan
Northwestern University
kedar.thiagarajan@northwestern.edu

Esteban Carisimo
Northwestern University
esteban.carisimo@northwestern.edu

Fabián E. Bustamante
Northwestern University
fabianb@northwestern.edu

ABSTRACT

Geolocating network devices is essential for various research areas. Yet, despite notable advancements, it continues to be one of the most challenging issues for experimentalists. An approach for geolocating that has proved effective is leveraging geolocating hints in PTR records associated with network devices. We argue that Large Language Models (LLMs), rather than humans, are better equipped to identify patterns in DNS PTR records, and significantly scale the coverage of tools like Hoiho. We introduce an approach that leverages LLMs to classify PTR records, and generate regular expressions for these classes, and hint-to-location mapping. We present preliminary results showing the applicability of using LLMs as a scalable approach to leverage PTR records for infrastructure geolocation.

CCS CONCEPTS

• **Computing Methodologies** → **Machine Learning**; • **Networks** → **Network measurement**.

KEYWORDS

Internet Measurement, Natural Language Processing, Large Language Models, Internet Geolocation, RIPEAtlas

ACM Reference Format:
Kedar Thiagarajan, Esteban Carisimo, and Fabián E. Bustamante. 2024. Poster: Revealing Hidden Secrets: Decoding DNS PTR records with Large Language Models. In *ACM SIGCOMM 2024 Conference (ACM SIGCOMM Posters and Demos '24), August 4–8, 2024, Sydney, NSW, Australia.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3672202.3673717

1 INTRODUCTION

Geolocating network devices is essential for various research areas (e.g., [3–5, 8, 13]). Yet, after two decades and despite notable advancements, it continues to be one of the most challenging issues for experimentalists [9].

Geolocating devices can be divided into two distinct problems: geolocating end-hosts and geolocating network infrastructure. While end-host geolocation has advanced significantly due to its commercial value, infrastructure geolocation remains significantly underdeveloped, and techniques commonly used for geolocating end-hosts do not always translate well to routers and servers. For instance, while latency-based geolocation is generally effective for end-hosts, routers often ignore ICMP echo requests.

An approach for geolocating infrastructure that has proved effective is leveraging geolocating hints in PTR records associated with network devices. Network operators encode physical location hints in DNS hostname strings of network devices to help with troubleshooting and operation [1] and previous work has shown the potential value of leveraging this information [2, 6, 10, 11].

As early as 2002, Rocketfuel [11] used manually-assembled collections of regular expressions (regexes) to extract PTR geolocation hints. Most recently, several efforts have tried to automate te task of extracting this location hints. The task of extracting and interpreting geo-hints from PTR records is challenging. For starters, the labels are primarily designed for human interpretation rather than computational processing. In addition, there is a lack of standardization across operators in what geographic information is encoded and how, which leads to the development of an ad-hoc approach for each codification. Even within a single operator, legacy infrastructure from rebranding and mergers and acquisitions results in multiple standards that can take decades to converge. For example, although the merger was executed almost 20 years ago, AT&T still uses South Bell Corporation Global labels, such as `99-170-164-205.lightspeed.tukrga.sbcglobal.net`. This often appears in networks with large geographic spans managed by multiple teams and divisions, as seen in companies like Google.

Huffaker et al. [2] tries to automate part of the task by searching for geographic encoding based on a previously populated dictionary of geographic-related strings. More recently, Luckie et al. [6] automatically extract and interpret geo-hints embedded into hostnames using regexes informed by a dictionary that includes strings such as airport codes, city, state and country names), and learn simple deviations from geohints such as prefix (e.g., "ash" for "Ashburn") and partial matches (e.g., "ftcollins" for "Fort Collins").

While highly effective, the coverage of these approaches and the associated tools and datasets is limited largely due to the challenge of scaling up some of the needed steps to create candidate regular expressions. Hoiho [6], the software component implementing Luckie et al. approach, resorts to the MAXMIND database for the large majority of IPs it cannot geolocate based on PTR records. For a CAIDA ITDK dataset (`itdk-2023-03`, using traces collected between 8-13 of March, 2023), it was able to extracted records from 0.041% of the IP addresses, although about half of the records have associated PTR record information.

Our work is based on the observation that Large Language Models (LLMs), rather than humans, may be better equipped to identify patterns in DNS PTR records and create extraction rules, offering a path to significantly scale the coverage of tools like Hoiho. Our approach uses LLM to (1) classify PTR records into distinct groups

ACM SIGCOMM Posters and Demos '24, August 4–8, 2024, Sydney, NSW, Australia

ACM ISBN 979-8-4007-0717-9/24/08
https://doi.org/10.1145/3672202.3673717

based on the structure and potential geographic hints, (2) generate regular expressions based on these classifications, identifying patterns and consistent naming conventions, and (3) map the identified classifications and regex patterns to geographic locations by linking encoded hints with actual place names.

The following paragraphs describe our approach and present some preliminary results.

2 APPROACH & DESIGN

The eruption of Large Language Models (LLMs), e.g., GPT-4, redefined automating information extraction (IE) tasks, including Named Entity Recognition (NER) for specialized fields, such as identifying network infrastructure encodings in our case. These LLMs leverage few-shot learning (FSL) [12] to learn from limited data, simplifying the development of new frameworks without re-training. We adopt this model to develop pipelines employing modern LLMs to learn example patterns and create extraction rules from limited cases.

Instead of a one-shot approach, we divide the process of generating regular expresions and geohints into multiple intermediate steps to maximize their precision. Our approach to decoding PTR records using a multi-step process involving three distinct LLMs. Each is specialized for a particular task, working with a subset of records from a given provider, as follows:

Classification We use LLM to categorize PTR records into classes. The prompts in this stage guide the model to accurately identify and label each record according to its class, considering various features and patterns within the data.

Regex Generation Following classification, we employ a regex generation LLM to create regular expressions extracted from the patterns observed in the classified records. The prompts for this model generate regex patterns that can match and extract relevant information from the records. This is critical for precise parsing and interpretation, as it allows the system to handle a diverse range of record formats and structures with high accuracy.

Hint Map Generation The final component uses LLM for hint map generation. This model correlates specific hints to geographic locations or other relevant attributes. The prompts are designed to produce mappings that enhance the accuracy of decoding network information. By providing context-specific hints, this step aids in interpreting complex data and improves the system's overall efficacy.

3 PRELIMINARY RESULTS

To evaluate our approach we selected a subset of 680 ASNs to analyze. We chose all large cloud providers, tier 1 ISPs (i.e., ISP without customer to provider relationships), the top 390 AS from APNICS Internet population data (together representing over 80% of the Internet population), and the top AS by APNICs AS internet population data per country. In total, the dataset includes 680 ASNs and 854,317,370 PTR records. From our classification model, we find that 23% of these ASNs encode geographic information with a finer granularity than country level in their records.

Table 1 summarizes the key parameters of this dataset and our mapping results. Our generated expressions and hint mappings cover 190 countries and identify 2,117 unique cities through 5,096 hints. Additionally, the analysis identified 1,409 unique regular expressions (regexes) used for geolocation.

Dataset	
# of ASNs	680
Frac. Eyeballs	81.82 %
# of Countries	190

Mapping results	
# of Hints	5096
# of Countries	190
# of Cities	2,117
# of regexes	1,409
Cost ($)	$120
Runtime	6 hours

Table 1: Evaluation dataset and mapping results

We apply our approach to a dataset containing 51,840 ASNs and 1,282,817,253 PTR records collected by OpenIntel [7]. Out of these, we generate regular expressions and hint mapping for 680 ASes.

First Attempts At Validation. We extracted geo-hints from AT&T (AS7018) records from CAIDA ITDK dataset (`itdk-2023-03`), a particularly challenging operator due to non-standard encodings. Hoiho extracts data from 563 out of 239,796 AT&T records. Our LLM-based approach is able to extract 38,883 records. Our validation involves pinging to geolocate AT&T devices from RIPE Atlas probes in the same city, a different city in the same country, a different country, and a different continent. Figure 1 shows clearly separated latency distributions, suggesting that the majority of devices were correctly geolocated.

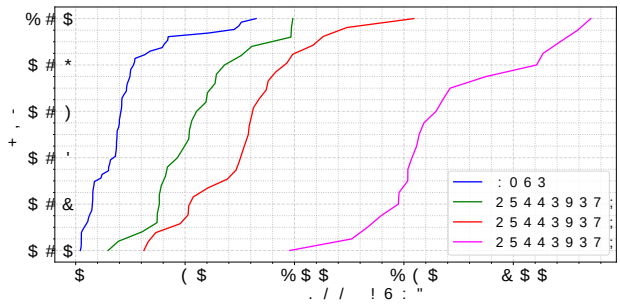

Figure 1: CDFs of RTTs from probes in a different distance from the geolocated device.

4 CONCLUSION

We make the case for an LLM-based approach to extract geo-hints for network devices, reducing the reliance on manual tasks of current approaches. We extract geographic information and perform an initial validation with records from AT&T, finding that our extracted geo-hints correspond to lower RTTs when using that information to select probes.

5 ACKNOWLEDGMENTS

The authors thank the anonymous reviewers for their valuable feedback. This work is supported by National Science Foundation grants CNS-2107392 and CNS-2246475.

REFERENCES

[1] Rahel A. Fainchtein1 and Micah Sherr. 2024. You can Find me Here: A Study of the Early Adoption of Geofeeds. In *Proc. of PAM*.

[2] Bradley Huffaker, Marina Fomenkov, and kc claffy. 2014. DRoP: DNS-based router positioning. *ACM SIGCOMM Computer Communication Review* 44, 3 (jul 2014).

[3] Costas Iordanou, Georgios Smaragdakis, Ingmar Poese, and Nikolas Lautaris. 2018. Tracing Cross Border Web Tracking. In *Proc. of IMC*.

[4] Josh Karlin, Stephanie Forrest, and Jennifer Rexford. 2009. Nation-state routing: Censorship, wiretapping, and BGP. (2009).

[5] Dave Levin, Yundo Lee, Luke Valenta, Zhihao Li, Victoria Lai, Cristian Lumezanu, Neil Spring, and Bobby Bhattacharjee. 2015. Alibi Routing. In *Proc. of ACM SIGCOMM*.

[6] Matthew Luckie, Bradley Huffaker, Alexander Marder, Zachary Bischof, Marianne Fletcher, and K Claffy. 2021. Learning to extract geographic information from internet router hostnames. In *Proc. of CoNEXT*.

[7] OpenINTEL. 2024. OpenINTEL rDNS Dataset. https://www.openintel.nl/data-access/

[8] Ramakrishna Padmanabhan, Aaron Schulman, Dave Levin, and Neil Spring. 2019. Residential links under the weather. (2019).

[9] Ingmar Poese, Steve Uhlig, Mohamed Ali Kaafar, Benoit Donnet, and Bamba Gueye. 2011. IP Geolocation Databases: Unreliable? 41, 2 (April 2011).

[10] Quirin Scheitle, Oliver Gasser, Patrick Sattler, and Georg Carle. 2017. HLOC: Hints-based geolocation leveraging multiple measurement frameworks. In *Proc. of TMA*.

[11] Neil Spring, Ratul Mahajan, and David Wetherall. 2002. Measuring ISP topologies with rocketfuel. In *Proceedings of the 2002 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications* (Pittsburgh, Pennsylvania, USA) *(SIGCOMM '02)*. Association for Computing Machinery, New York, NY, USA, 133–145. https://doi.org/10.1145/633025.633039

[12] Yaqing Wang, Quanming Yao, James T Kwok, and Lionel M Ni. 2020. Generalizing from a few examples: A survey on few-shot learning. *ACM computing surveys (csur)* 53, 3 (2020), 1–34.

[13] Z. Weinberg, S. Cho, N. Christin, Vyas Sekar, and Phillipa Gill. 2018. How to Catch when Proxies Lie: Verifying the Physical Locations of Network Proxies with Active Geolocation. In *Proc. of IMC*.