Effectiveness of Live-TV content caching on content delivery networks

Tshifaro Justin Ndivhuwo(SUQG0J)

Supervisor: Mr Adolf Kamuzora

June 10, 2022

Acknowledgement

I have to express out appreciation to the Mr Adolf Kamuzora for sharing his pearls of wisdom with me during the course of this research.

Abstract

The goal is to analyze the performance of deployed CDN caching system by deriving performance metrics and show how a properly chosen metric can indicate if the cache is properly utilized (quality of experience, cache efficiency, ingress/egress throughput, content popularity, etc)

Content Delivery NetworkSupervised and Unsupervised learning.

1 Introduction

A content delivery network simply known as CDN is a group of geographically distributed servers that speed up web content delivery by placing it closer to the location of users. These networks eliminate the traffic bottleneck that can occur when serving content with a single server, by distributing text, image, and video data to edge sites throughout the world. CDN is important for Internet service providers as they are constantly striving to improve the user experience (QoE).They were developed to resolve network congestion caused by the delivery of web content.

Caching is a process of storing data so that future requests for that data can be served faster. Catching is the heart of Content Delivery Network(CDN) services because CDNs cache content like web pages, images, and video in proxy servers near to physical location of users.In this experiment we will study the effectiveness of Live-TV content caching on content delivery networks.One of the most important metric in this experiment is the cache hit ratio, a metric that compares how many content requests a cache can successfully serve from its cache storage versus how many requests it receives. A high cache hit ratio indicates a good-performing CDN.

1.1 How does a CDN work?

The main goal of a CDN is to reduce latency, the time taken for data to get to its destination across the network. Low latency is linked to a positive user experience, whereas excessive latency is linked to a negative. Most Content Delivery Networks try to resolve latency problem by reducing the physical distance between the servers and the users(time traveled). For example, when one is trying to stream a live video, the CDN finds an optimal server on its network to serve up that video. Usually, that will be the server closest to the user's physical location. The video files are cached and will remain on the same server for other user's that will request the same video in the same geographic location. The server is constantly updated with new content to serve any future request.

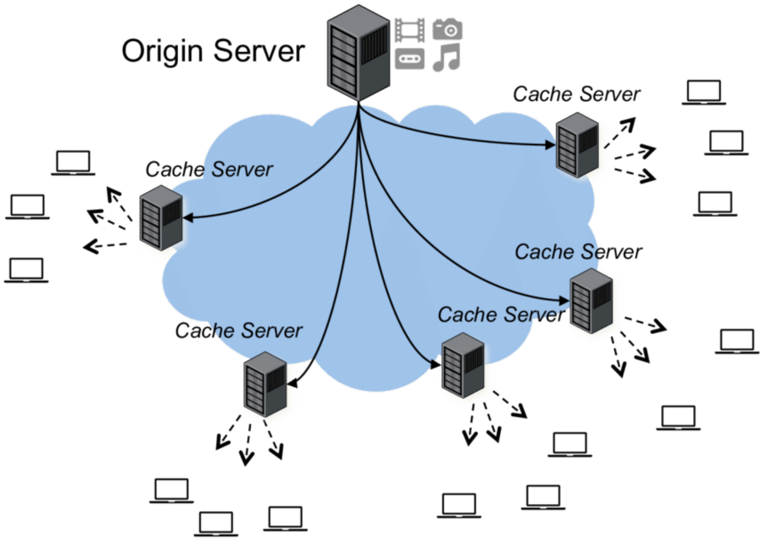

Figure 1: Content Delivery Network architecture

1.2 Why CDNs are important for live streaming

When users join a live stream, it is easy to deliver content to small number of viewers but it becomes difficult to deliver contents to a large number of viewers from all over the world. CDNs brings the live-streamed content closer to the viewers which results in lower load on the origin servers. This will also results in a lower latency.

1.3 CDN Logs

To transport video across the internet to our homes and mobile devices, CDNs use a number of log files known as CDN logs. The performance of the CDN servers and the quality of video streaming are also recorded in these logs. A

CDN log provide useful information, for example, CDN logs will tell you how many times your site has been visited, which domains are the most frequent visitors, and which files have been downloaded.Most importantly, CDN logs can give CDN providers and their customers useful information regarding the quality of their CDN services. Below is an example of a CDN log

```
127.0.0.1 username [10/Oct/2021:13:55:36 +0000] “GET /my_image.gif
HTTP/2.0” 200 150 1289
```

1. **IP Address (127.0.0.1):** The user IP source from which the data was requested. It can give service providers an information on how someone is using their site.

2. **Username (username):** Some providers will try to decode the authorization header in the incoming request to determine the username. The username and password are encoded in a basic authentication request, for example. If you see any questionable activity, you may be able to connect it to a specific account and disable it.

3. **Timestamp (02/JAN/2022:13:56:35 +0000):**This section of the log shows when we made a request. When showing this data on graphs, it's one of the most common values.

4. **Request Line ("GET /my_image.gif HTTP/2.0"):**The application and item the user requested are indicated by the HTTP GET and POST statuses used in this query.

5. **HTTP Status (200):**A general-purpose return value is a status code followed by a 2. It is used to confirm response success. Depending on the number appended to the return code's conclusion, the same code implies a different result.

6. **Latency (159):**Latency is an extremely significant indicator and metric to monitor. There's a potential that your end users will notice a slowdown in response when latency fluctuates. Slight reductions in latency suggest that the CDN is in good working order.

7. **Response size (1388):**Understanding the response size might assist you in determining whether or not your application is overburdened.

It is important for service providers to understand the Logs. In this experiment, Logs are some of the metrics that will be used.

2 Data cleaning and processing

Data processing is important in data mining as it provides better results that are, more accurate and reliable. The aim is to organise the data and covert it into its best organised form. Understanding the data set is one of the most important steps to be executed.We have the VoD Data set that has about 1577063 instances and about 22 attributes. The data set had few missing values or NaN(Not a Number) values, that were replaced by the most frequent value, which we will have minimal to no effect on our results.There are columns that were dropped because they serve no purpose or they are not important for our experiment, e.g device-brand.The "time_received" was in a string type, we had to convert it to pandas date-timee' type in order for us to analyse it. The table below shows the detailed description of the data set.

Feature	Description
statuscode	HTTP response status codes
content_type	Indicates the media type of the resource
request_method	Http version (protocol)
byte_sent	The size of the resource, in decimal number of bytes
time_to_serve	time needed to process the request
session_id	id uniq to a streaming session
cachecontrol	Directives for caching mechanisms in both requests and responses
request_header	User-Agent'version, eg. browser's version
user_agent	User-Agent,usually client's app
device_type	Type of the device
request_url	URL (address of request)
time_received	Arrival time of the request
vod_id	VoD asset identifier
user_loc	Long. and lat. of the client based on geoip lookup
live_tv_id	Live TV channel name
devicebrand	Client's device brand
host_id	Specifies the domain name of the server (for virtual hosting)
method	Http request method eg. Get,post
vod_encoding	VoD asset encoding version
response_status	HTTP request was a cache hit or miss
user_id	id unique to a single user

Table 1: List of log line features

3 Methodology and Analysis

There are important metrics that we need to analyse to get reliable results. The data set contains caching system logs. There logs contain important metrics we need. When the cache starts, log files are created and are kept until the cache is stopped.The full report of each user's requests is included in logs, together with the result of the caching operation and the time required to complete each request.

The following data was recorded for each request:

1. time-received (millisecond)
2. time-to-serve (microsecond)
3. user-id
4. response-status
5. bytes-sent
6. method
7. request-url
8. request-header

Some of the features are not useful to our experiment, we will not discuss them at this point.

The first metric, time-received which is saved in milliseconds reflects the time when a request was made. Because the time is in milliseconds, there was not much difference in time and some of the request were recorded at the same time. This will make analyzing the time between successive queries more difficult.

The system's time-to-serve is the amount of time it takes to complete a request.Time to process the request, locate it, and construct a response. The request might be locally or remote. Again this is a very a low time resolution.Logically, one would think that requests served from the local cache are likely to be processed faster than those requiring a remote retrieval. However, as we will see in the following, this is not always the case.

The user-id, which is unique to a single user. In this case, this will be equivalent to the The user's IP address that made the request. If the request includes an X-Forwarded-For header, the user's IP is chosen from the same pool. We found that some id's made numerous requests.Many client types were logged with the same IP address (multiple client types can be detected by looking at the user agent).This could indicate that the consumers have many devices(Like Phones and tablets) in their home. The response status indicates the type of request and whether or not the cache was able to locate the item locally. In this experiment, the response can either be a TCP_HIT or TCP_MISS. This is one of the most important metric as we are going to need it to get the hit/miss ratio.Hit and miss ratios are important because they can tell you how well your cache is operating and whether it has been tuned. Request methods are defined in HTTP to express the desired action for a specific resource.It's important for VoD resources accessed over HTTP. We discovered that nearly majority of the VoD queries came from HTTP clients.

The request url is the URL of the stream or site. In our case, the format is different(123458) and we assumed that this number represent the url_id. The URL can be used to link requests arriving from the same IP address. These requests will be referred to as a flow because they represent pieces of an audio/video stream being played by the user.

User agent Identifies or describes the software agent that submitted the request. The value that will be passed to an origin server varies according to request type.In simple terms, the browser that is used by the client. In our case we found that different browser have different behaviours as some can be slow and some can be faster. Our observation was mostly based on the version of the browser.

4 Results

The log data was processed in python using pandas, which is a popular open source Python module for data science, data analysis, and machine learning tasks.Pandas is built on top of Numpy, a library that supports multi-dimensional arrays. Matplotlib was used for visualization, it is a comprehensive library for creating static, animated, and interactive visualizations in Python

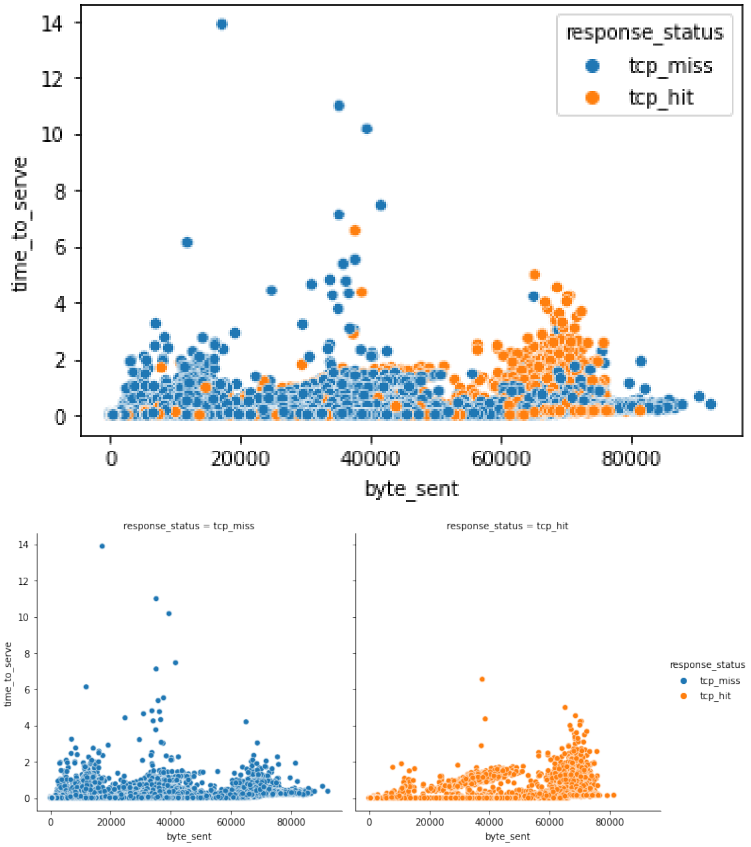

Figure 2: Time to serve vs size

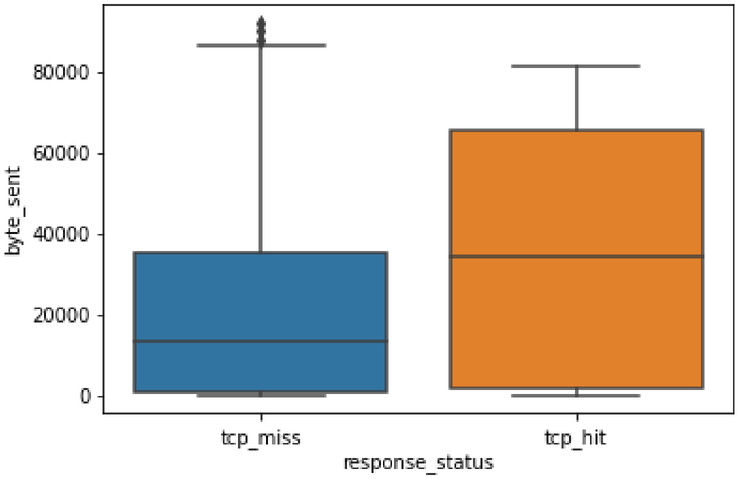

Figure 3: Throughput for the hit and miss requests

The initial assessment was on the consumers' perceptions of Quality of Experience (QoE or QoX). The time it takes to fulfill a request might be used as a very rough indicator.It is wise to keep in mind that the requirement in VoD systems is that the next block be transferred before the preceding one's playout is finished. As a result, the QoE is best if the time to serve is bounded rather than the shortest possible.

We anticipate that serving a request from the local cache will take less time than serving a request that must be retrieved from the acquirer.In our case this is not evident as observed is in Figure 2.

Figure 3, illustrates that hit requests have a better throughput than missed requests. The low throughput could indicate that customers are just tapping into a small portion of the CDN's capabilities, and that a spike in traffic is imminent.

The hit ratio is a widely used metric for determining cache effectiveness. In theory, the higher the number of requests served by the cache (hit), the better. The hit ratio can be misleading when it come to judging the effectiveness of cache performances. For this experiment, we only calculated the , **1),the raw hit ratio**, which is evaluated by Counting the number of hits vs. the number of requests. After analysing the response status, the following results were found: tcp_mis = 945557 tcp_hit = 631506. The overall or raw hit ratio was only 40% which is very low for a CDN to performance.This is obviously because there were more miss that hits.The cache hit ratio isn't the be-all and end-all of CDN performance, other elements play a significant role in determining a CDN's success. It's also vital to consider where the content is served from. A CDN should, in theory, offer material from the CDN server that is nearest to the end user. If this does not happen, the CDN's performance will suffer.

We isolated the inter-arrival time of the queries to further evaluate the situation.

The inter-arrival time is already indicated by the resource request density (Figure 4) (notice that the ordinate is in logarithmic scale). It is also wise to notice that the inter-arrival starts getting high, which means users would experience latency problems. There more the, the slower the network would be.

In order to further confirm this clue,we have drawn the Fitted distribution graph on inter-arrival in Figure 5. Also this test leads to the conclusion that the distribution is well approximated by uniform, and it is not short range dependant or limited.

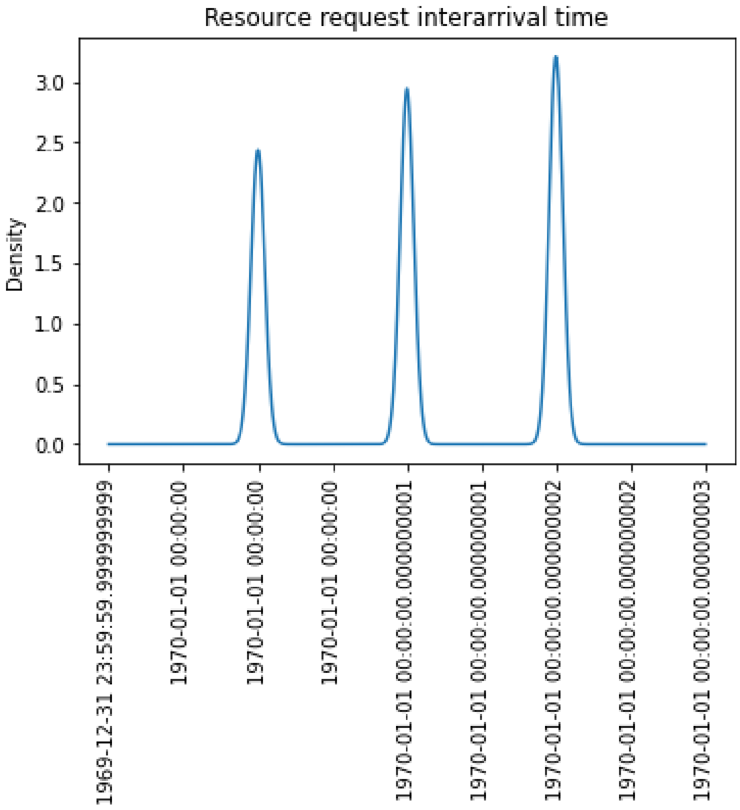

Figure 4: Resource request inter-arrival time

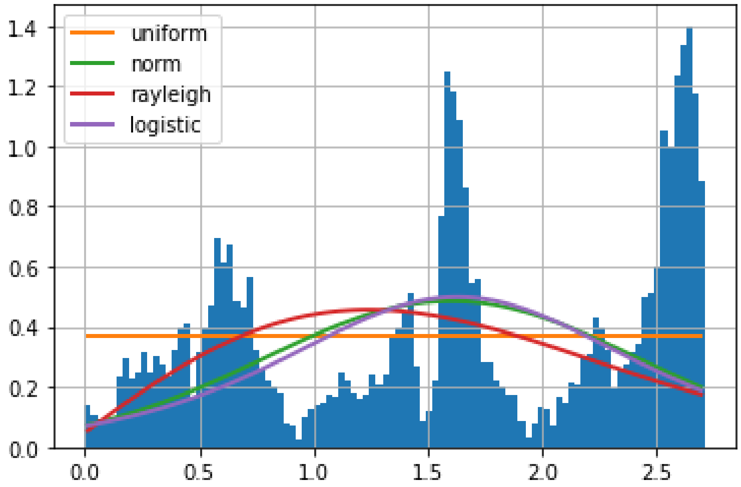

Figure 5: Distribution for the requests inter-arrival time

5 Conclusion

We investigated the performance of a real-world VoD caching system in this work. The findings revealed several unexpected behaviors.

we found that the hit ratio was too low due to the number of users and we can conclude highlight if the cache is optimal or it is serving a too small number of user, although we had shown that caches with a big user base should be favoured. Also we can conclude that time to serve and time to receive data must be optimal for the CDN optimal performance

In the instance of VoD requests, we have shown that the user traffic pattern follows an LRD distribution. The performance restrictions of the CDN caches are set by this element. These two major characteristics should be considered throughout the CDN design phase in the future.

References

[1] Itequus Homepage , https://www.intequus.com/cdn-metrics/ . 28 Aug 2019

[2] Emir Beganović , https://blog.apnic.net/2019/08/28/analysing-global-cdn-performance/. 28 Aug 2019

[3] IBM Cloud Education, https://www.ibm.com/cloud/learn/content-delivery-networks. 23 December 2020

[4] LNCS Homepage, https://www.datainsightonline.com/post/how-to-find-the-distribution-that-fits-your-data-best. Last accessed 4 9 Nov 2021

GitHub Link: https://github.com/justin9503/Data-Science-Lab-1