



GRAN SASSO
SCIENCE INSTITUTE

On Device vs Remote LLMs

Consequences of network quality

*Kyryl Veremiov
Manuel Antonio Vilas Valiente*

www.gssi.it      

Road map

1. Replicate as much as possible the experiments described in the paper *“On-Device or Remote? On the Energy Efficiency of Fetching LLM-Generated Content”*
2. Design and execute the experiment with the 3 levels of network connection quality (normal, medium, bad)
3. Record the energy consumption of different computer components involved in the experiments
4. Perform an analysis of energy consumption behavior in each of the designed scenarios

Experiments

Topics:

- United States
- Donald Trump
- Elizabeth II
- India
- Barack Obama
- Cristiano Ronaldo
- World War II
- United Kingdom
- Michael Jackson
- Elon Musk

Sizes:

- short: 100
- medium: 500
- long: 1000

Template:

"In **{size}** words, please give me information about **{topic}**."

Experiments

Softwares:

- LM Studio 0.3.37
- Python 3.12.9
- HWiNFO 64 v8.40-5900

Models:

- liquid/lfm2.5-1.2b
- qwen/qwen3-4b-thinking-2507
- llama3.1-8b-chinese-chat
- openai/gpt-oss-20b

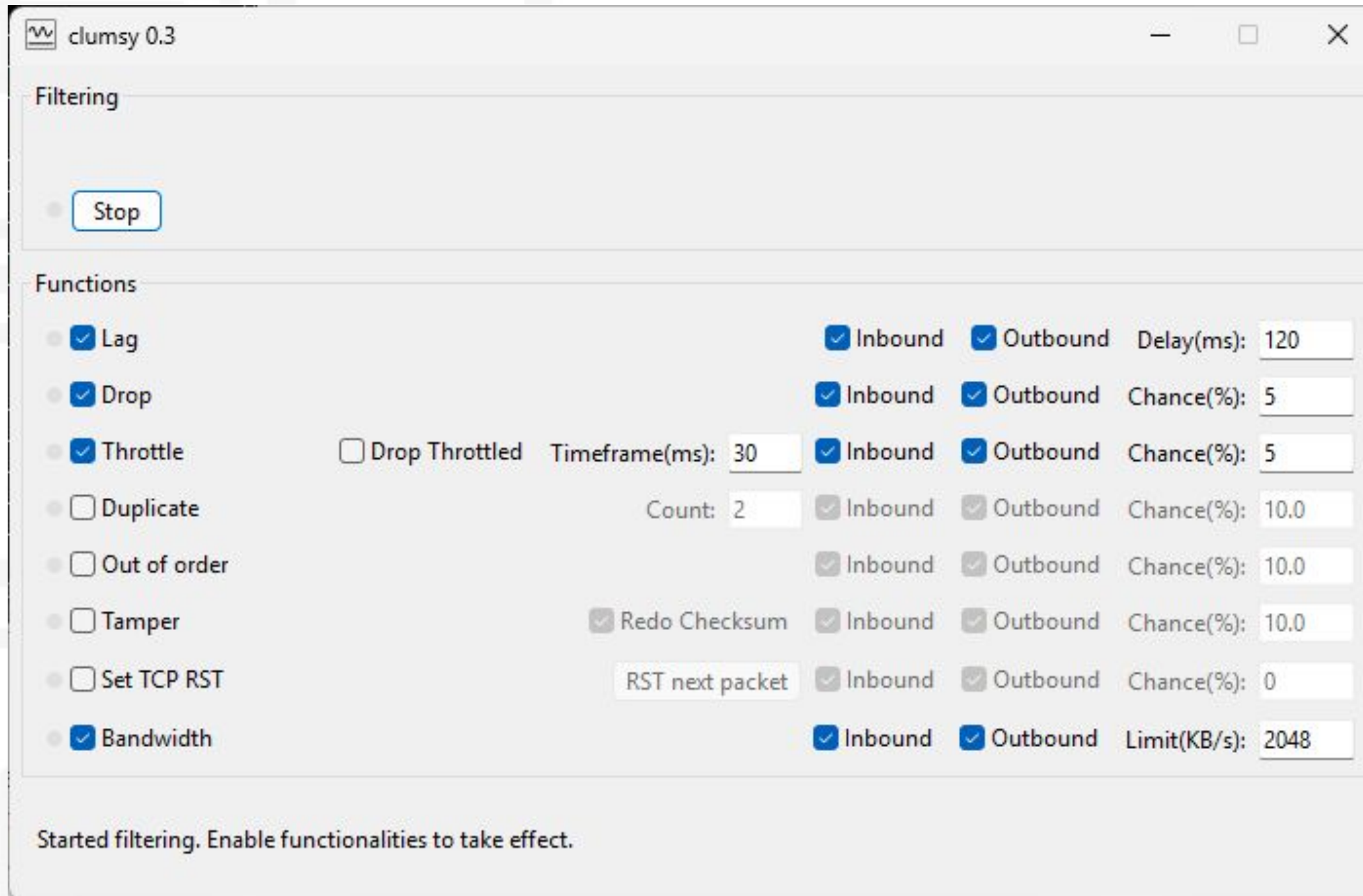
Client

- Acer Nitro AN515-55
- Windows 11
- CPU Intel Core i5-10300H
- GPU NVIDIA GeForce GTX 1650
- RAM 24 GB DDR4

Server

- ASUS V16 V3607 Notebook Gaming
- Windows 11
- CPU Intel Core 7 240H
- GPU NVIDIA GeForce RTX 5060
- RAM 16 GB DD5

Experiments



Network Conditions:

- Normal
- Medium:
 - Lag: 120
 - Drop: 5
 - Throttle:
 - Timeframe: 30
 - Chance: 5
 - Bandwidth: 2048
- Bad:
 - Lag: 300
 - Drop: 10
 - Throttle:
 - Timeframe: 40
 - Chance: 10
 - Bandwidth: 128

Experiments

Normal



Medium



Bad



Experiments

- Topics: 10
- Repetitions per topic: 3
- Sizes: 3
- Models: 4
- Conditions: 5
 - Locally on Server
 - Locally on Client
 - Online with Bad, Medium and Normal network

TOTAL: $10 \times 3 \times 3 \times 4 \times 5 = 1800$

Results:

- **20 JSON files** with information about each request (times, number of tokens, responses, features)
- **5 CSV files** containing information from a large number of computer sensors (not just energy consumption)

Results

server_result_mean

	Virtual Memory Load [%]	Physical Memory Load [%]	Total CPU Usage [%]	GPU Total Usage [%]	Total Activity [%]	GPU Power [W]	CPU Package Power [W]	GPU Clock [MHz]	Core Clocks (avg) [MHz]	GPU Temperature [°C]	...	GPU energy [Joule]	CPU Package Power energy [Joule]	Total energy [Joule]	CPU energy [Joule]	Efficiency [number_of_tokens/Joule]
size_category																
long_bad	54.307500	68.294167	11.450000	3.120833	1.170000	18.773083	21.526700	549.933333	2075.596667	66.363333	...	633.203243	1008.154761	1641.358004	1008.154761	2.124292
long_local	65.929167	79.340833	18.275000	1.046667	2.550833	36.615592	23.122542	1095.091667	2731.909167	67.325000	...	1085.346312	1184.480005	2269.826316	1184.480005	1.502113
long_medium	56.618333	83.712500	22.114167	3.698333	5.793333	22.858850	28.547608	756.083333	2983.620833	69.688333	...	703.241007	1292.931381	1996.172387	1292.931381	1.595151
long_normal	58.207500	84.161667	21.042500	3.880000	7.090833	34.801750	25.333558	1253.125000	3407.500833	73.135833	...	899.407874	1128.470770	2027.878644	1128.470770	1.410445
medium_bad	54.321667	68.357500	11.467500	2.923333	1.080833	16.220225	20.090575	436.141667	2006.176667	63.735833	...	323.523308	554.616878	878.140186	554.616878	2.282586
medium_local	65.937500	79.574167	17.427500	1.115833	1.830000	35.450517	22.489550	1206.841667	2664.485000	65.638333	...	617.723611	613.563855	1231.287466	613.563855	1.494905
medium_medium	56.554167	83.592500	20.799167	3.690833	5.486667	24.042733	28.227292	764.133333	2918.193333	68.680833	...	405.256759	688.860015	1094.116774	688.860015	1.675689
medium_normal	58.195000	84.035833	20.697500	3.995000	7.367500	36.733108	24.657892	1409.983333	3340.658333	72.575000	...	540.561836	560.612703	1101.174539	560.612703	1.422930
short_bad	54.335000	68.335833	11.391667	2.895833	0.805000	15.952042	18.432875	459.150000	2038.595833	64.326667	...	94.425177	129.839113	224.264290	129.839113	1.664923
short_local	65.970833	79.617500	16.602500	1.218333	2.964167	31.754183	20.954517	1144.308333	2649.848333	66.170833	...	177.052552	157.275996	334.328547	157.275996	1.029863
short_medium	56.539167	83.567500	20.972500	3.572500	4.830833	23.197300	26.331925	716.733333	2966.910833	68.615833	...	100.524277	154.171383	254.695660	154.171383	1.435369
short_normal	58.235833	84.081667	22.495833	3.975833	11.773333	30.289458	24.414517	1045.291667	3305.835833	71.677500	...	115.369915	118.284560	233.654475	118.284560	2.174705

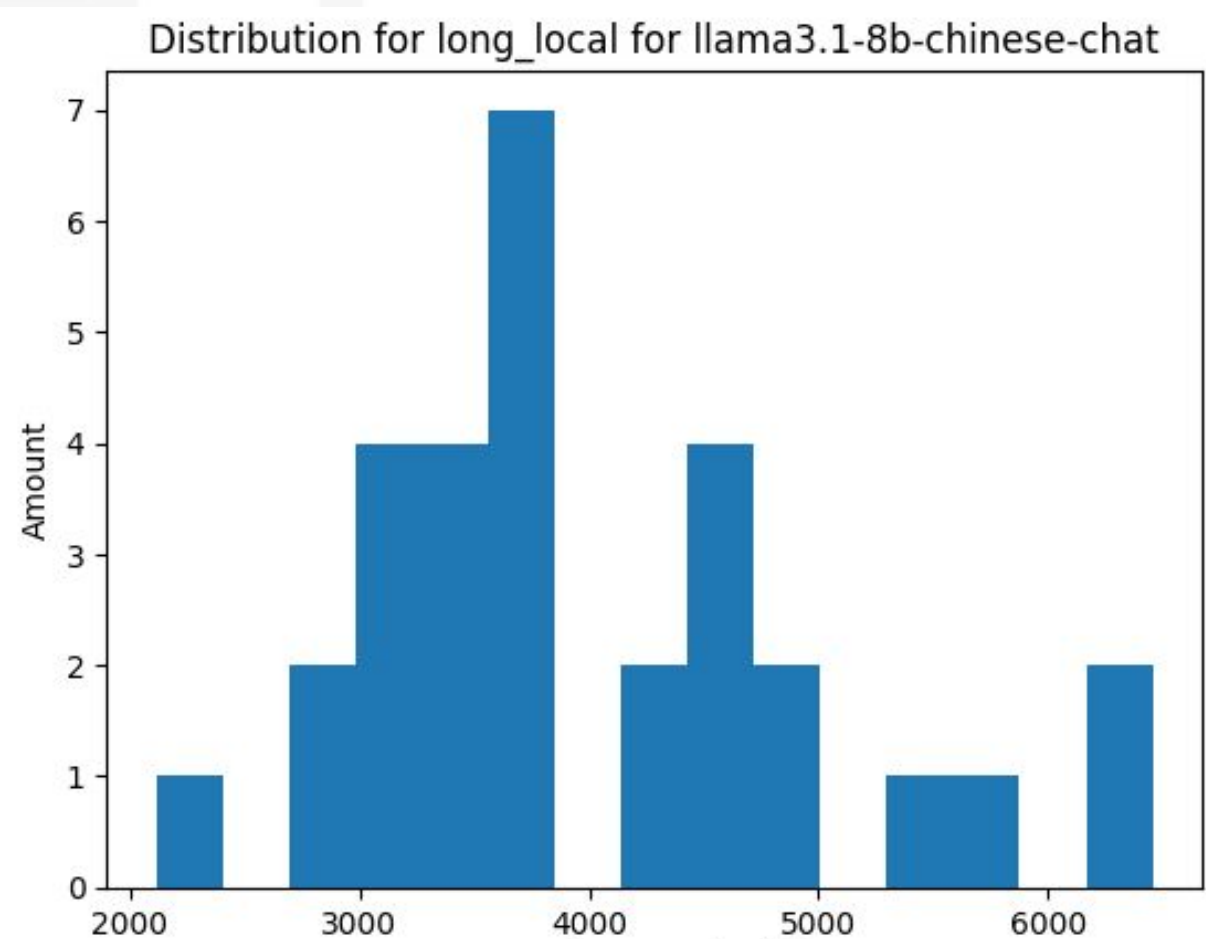
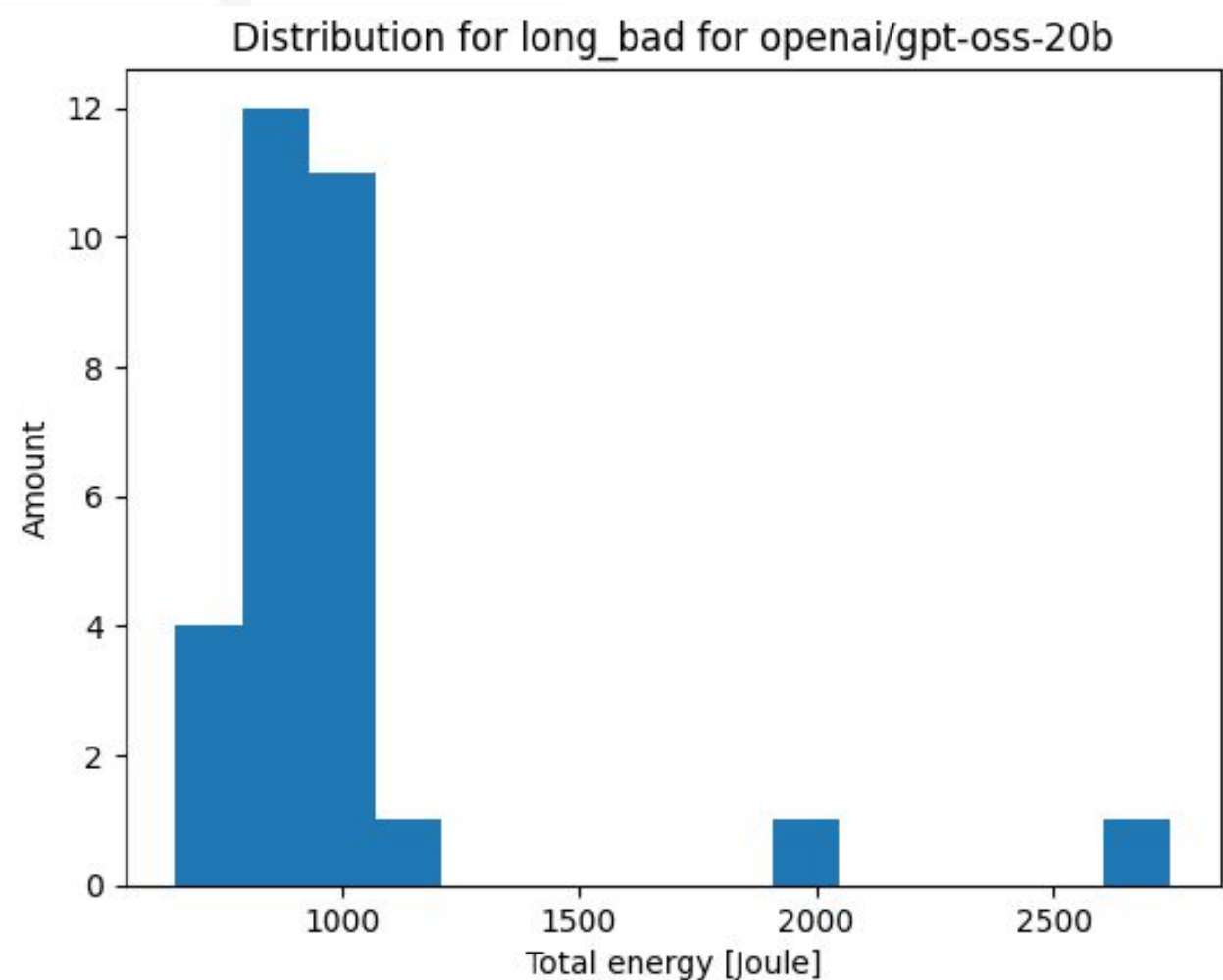
12 rows × 28 columns

Energy [Joule]= Power [Watt] * Time [second]

Efficiency [1/Joule]=Number_of_tokens/Energy



Distribution for query types



Total energy [Joule] for openai/gpt-oss-20b

	mean	std
size_category		
long_bad	1009.432470	391.195849
long_local	12759.338929	1249.721981
long_medium	901.009061	125.413286
long_normal	919.765106	96.514067
medium_bad	477.889393	84.729656
medium_local	6622.047583	518.875035
medium_medium	432.903520	31.650940
medium_normal	457.083031	31.418756
short_bad	110.957790	14.815975
short_local	1375.024214	250.163504
short_medium	98.495595	32.103927
short_normal	92.495391	12.815394

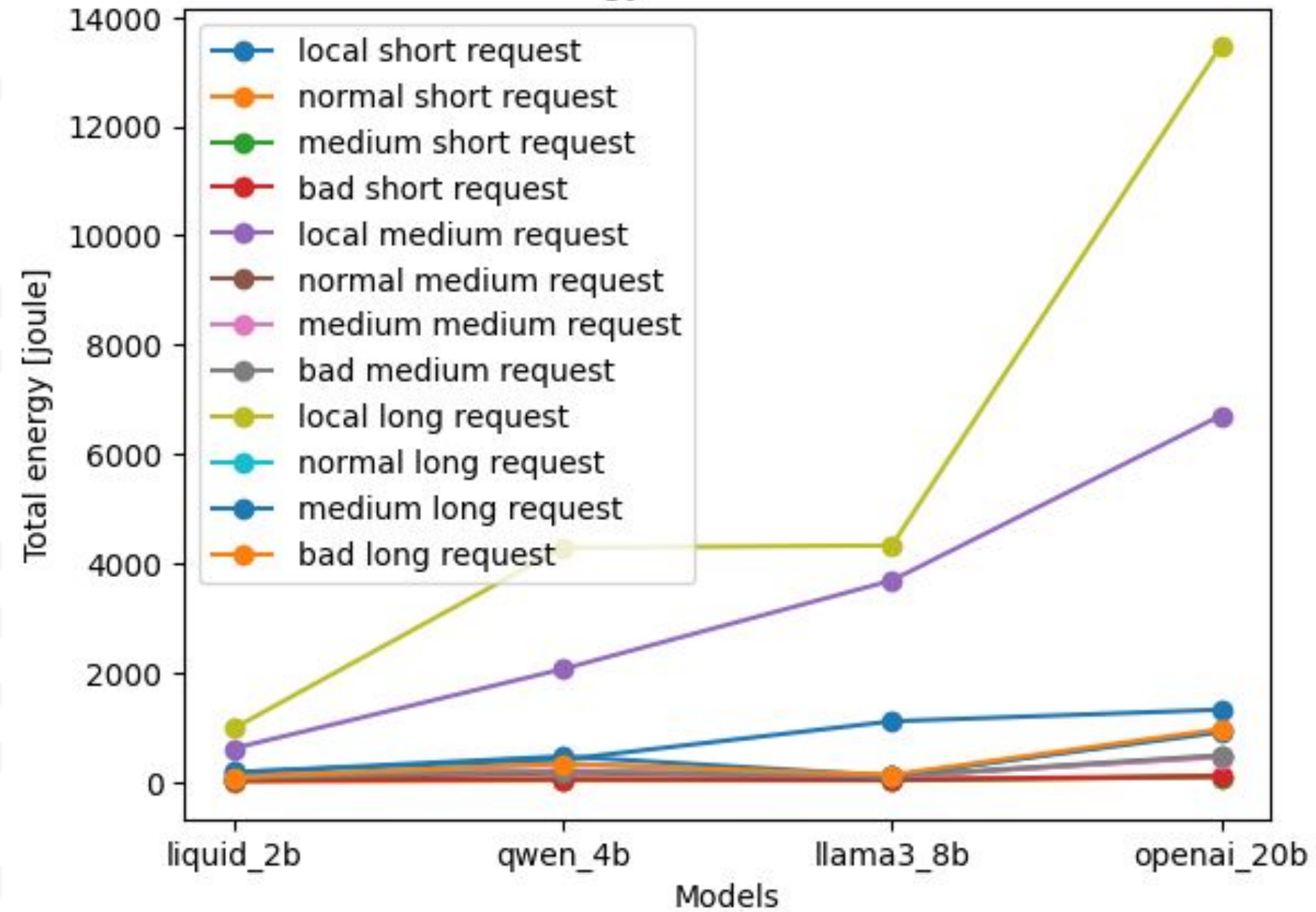
Total energy [Joule] for llama3.1-8b-chinese-chat

	mean	std
size_category		
long_bad	134.282332	77.998540
long_local	3995.620430	1018.020724
long_medium	104.142221	28.375181
long_normal	94.159376	29.340187
medium_bad	104.908773	23.700788
medium_local	3471.199781	840.045433
medium_medium	106.569187	51.556781
medium_normal	93.124836	33.313304
short_bad	47.527258	15.234440
short_local	1113.816894	239.629228
short_medium	38.844682	9.582146
short_normal	31.840509	9.576466

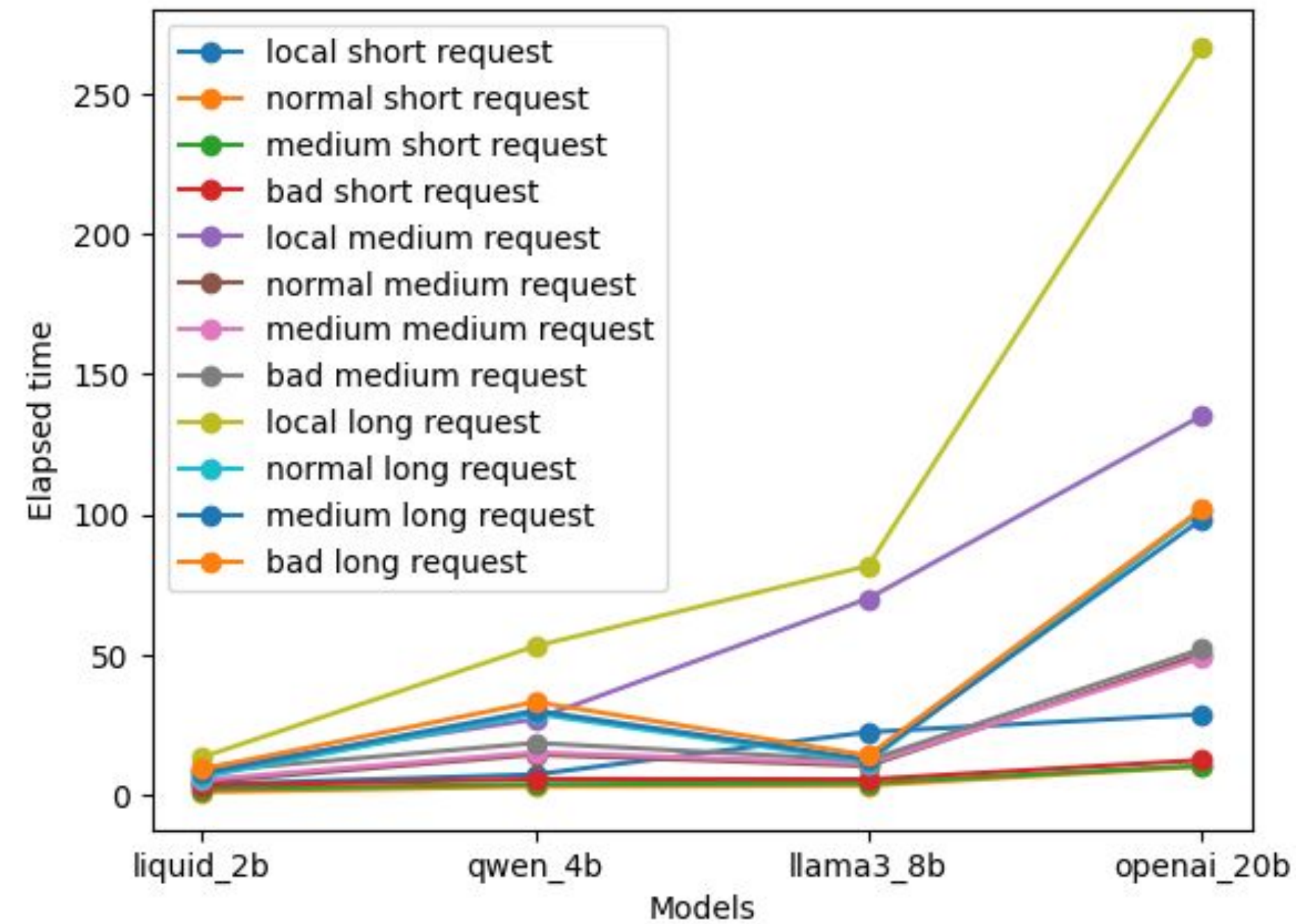


Considering different models

Energy result for client

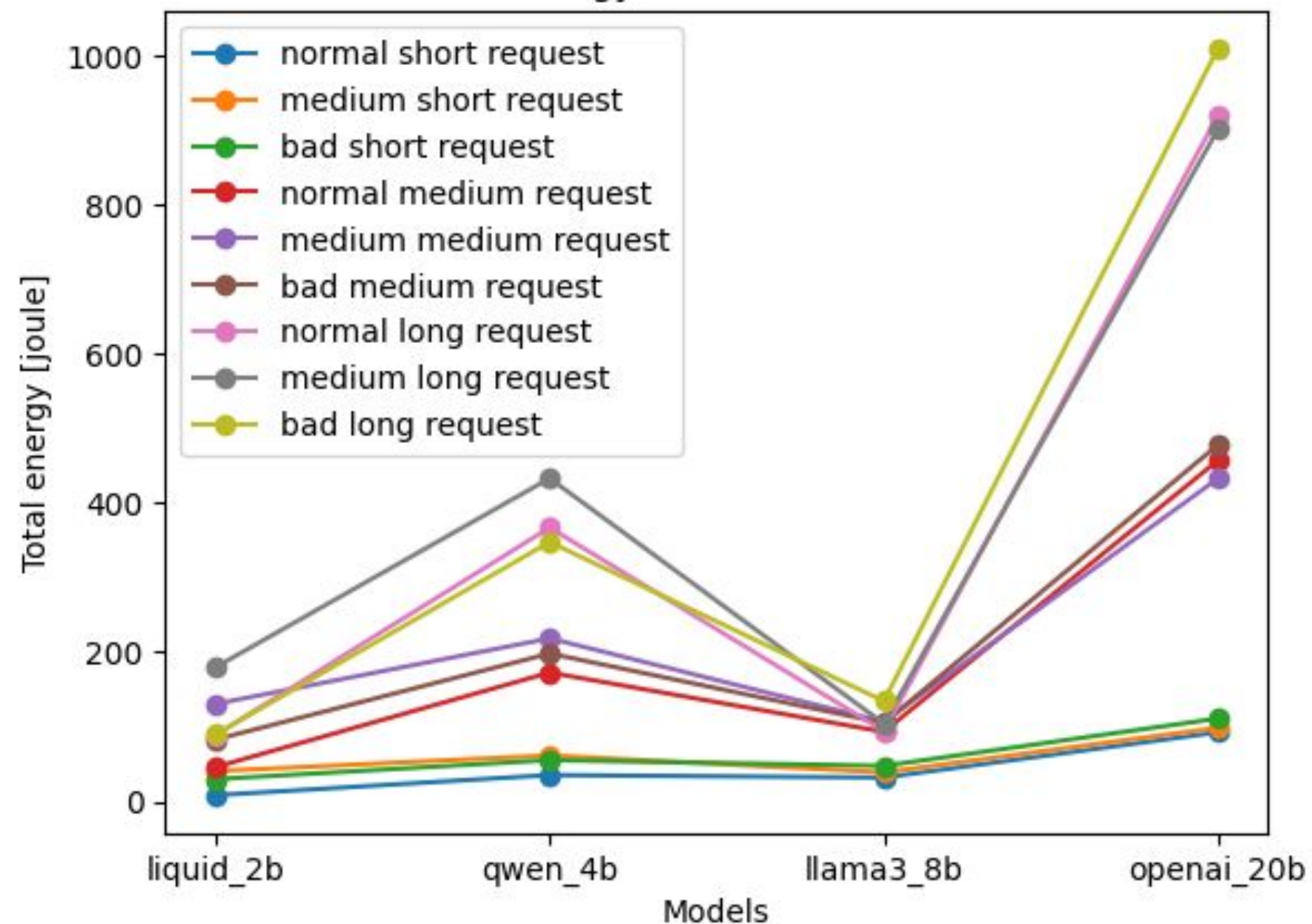


Time result for client

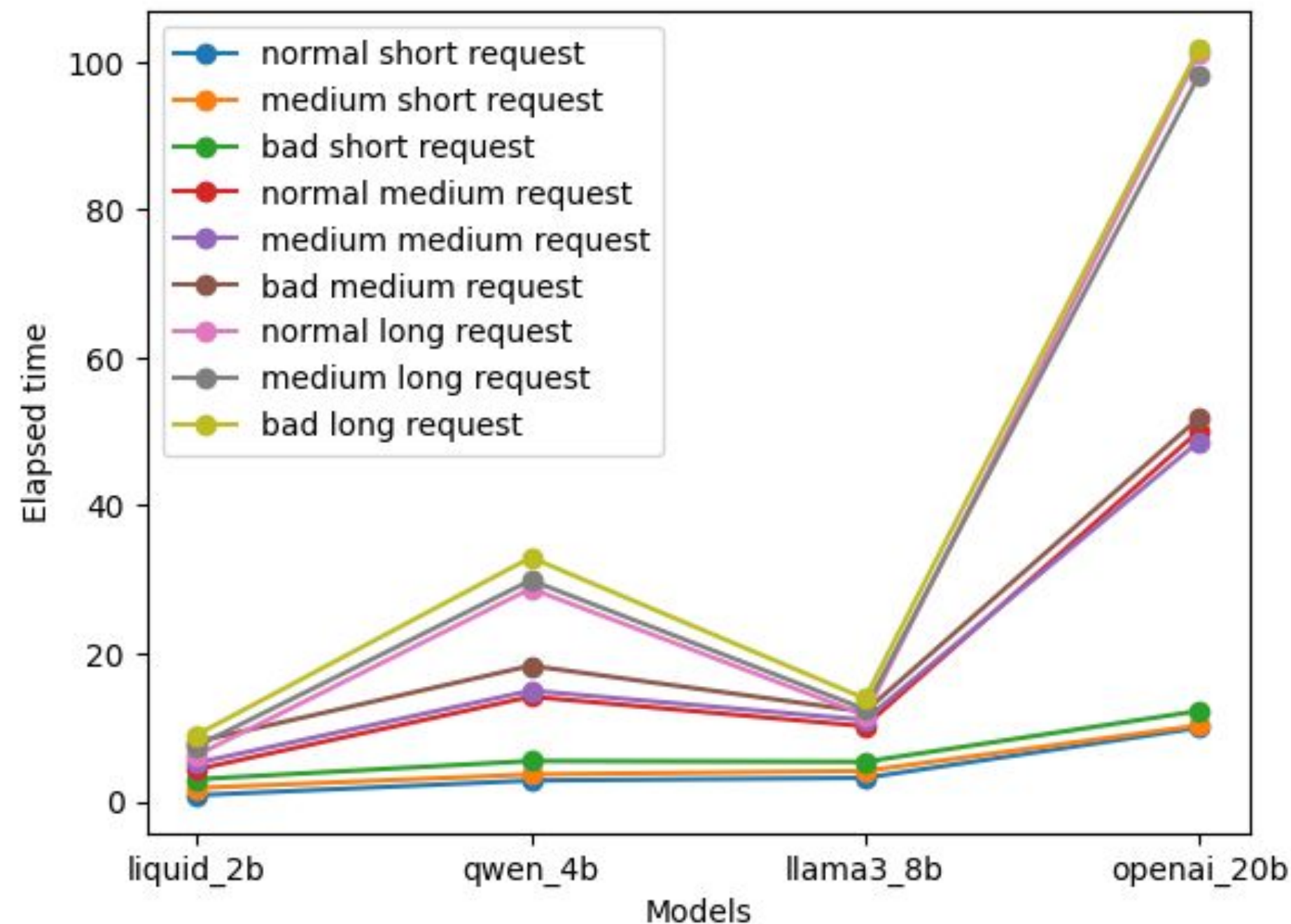


Considering different models

Energy result for client

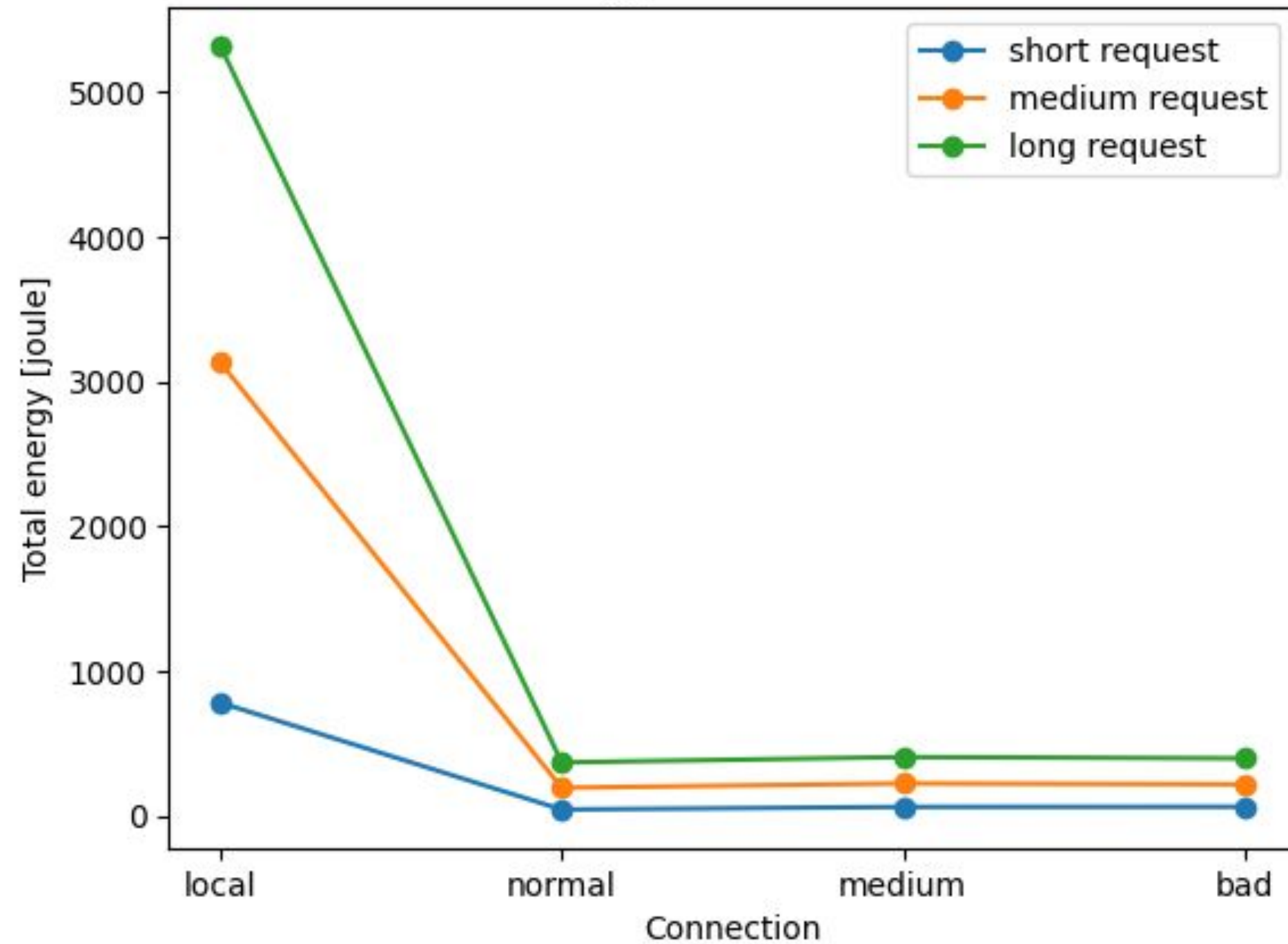


Time result for client

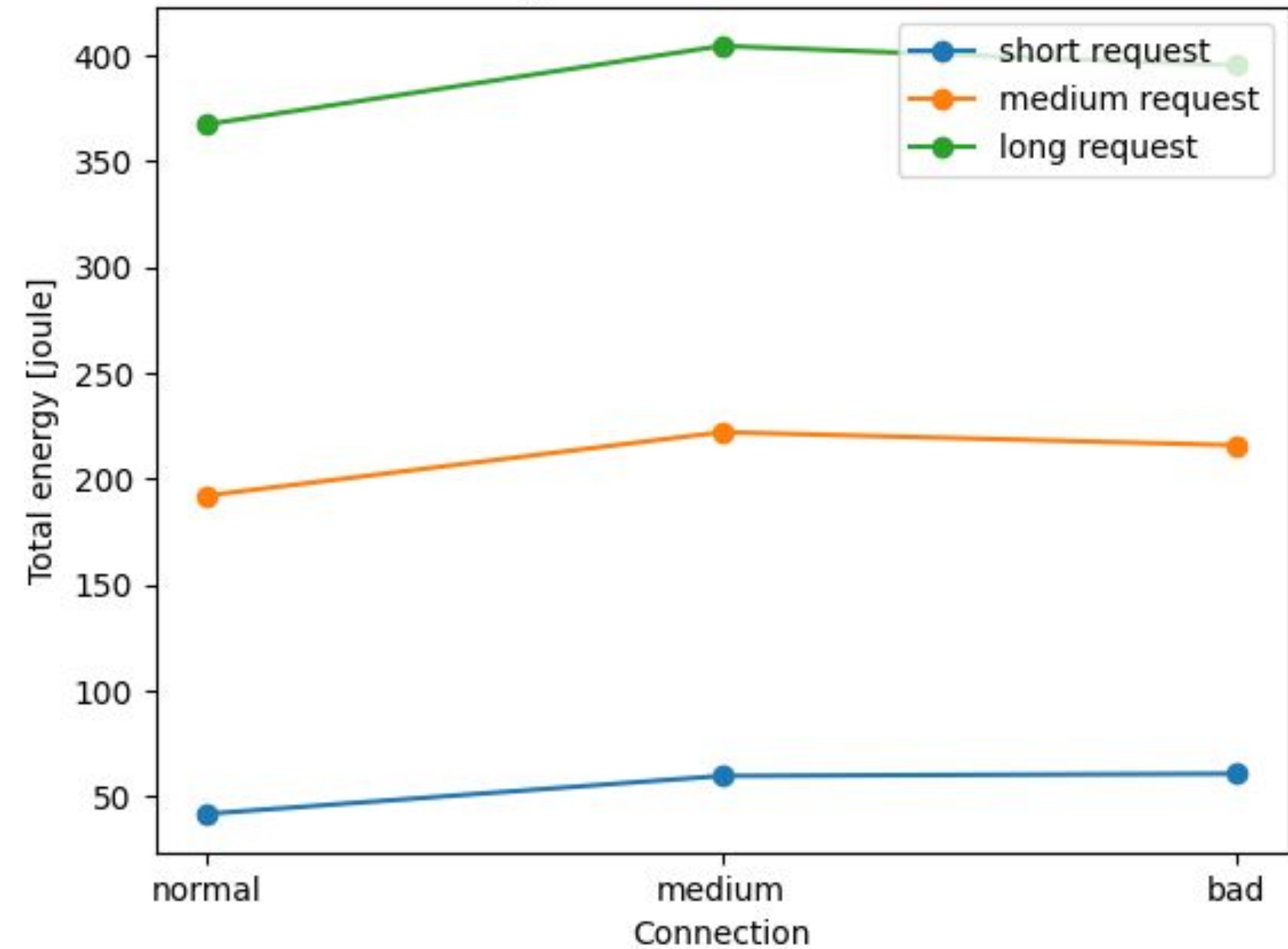


Considering various connection

Energy result for client

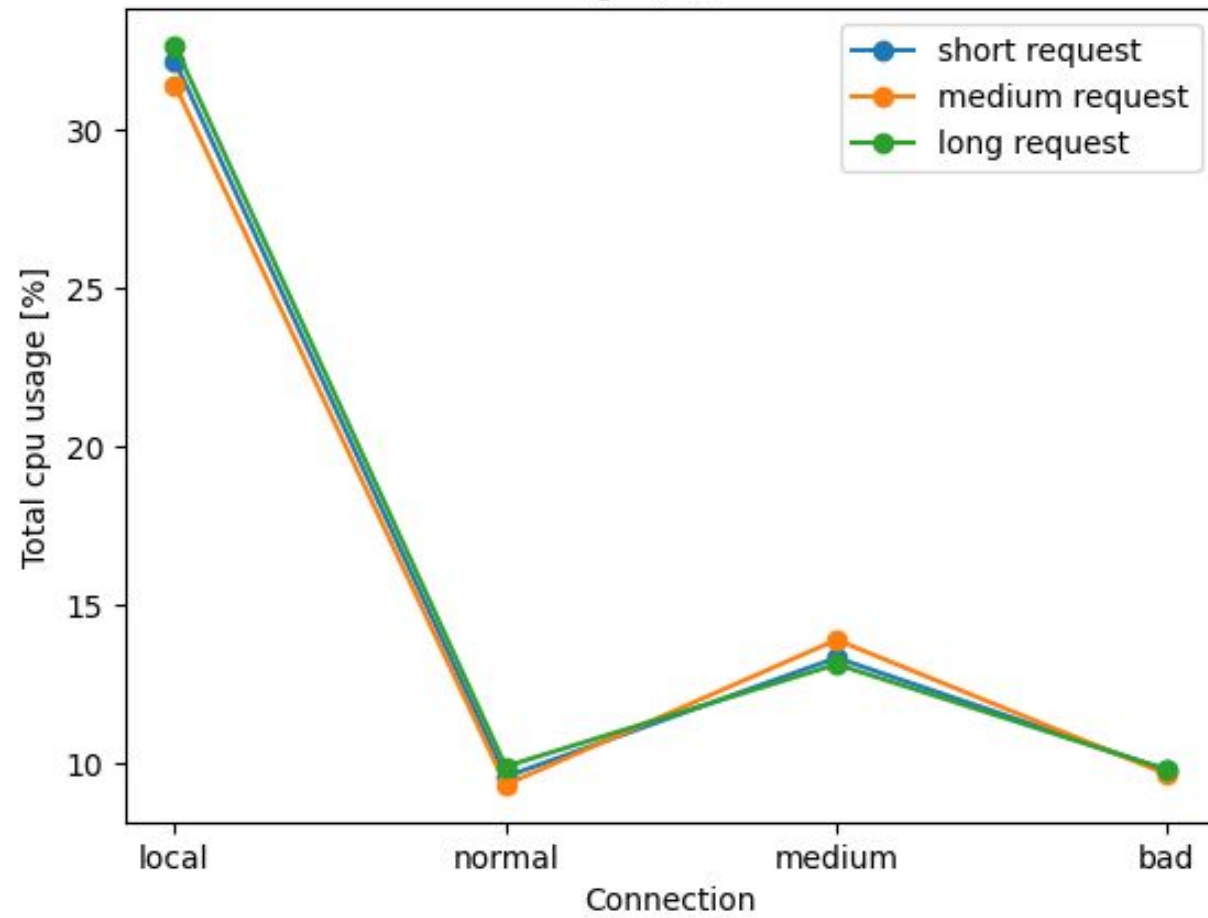


Energy remote result for client

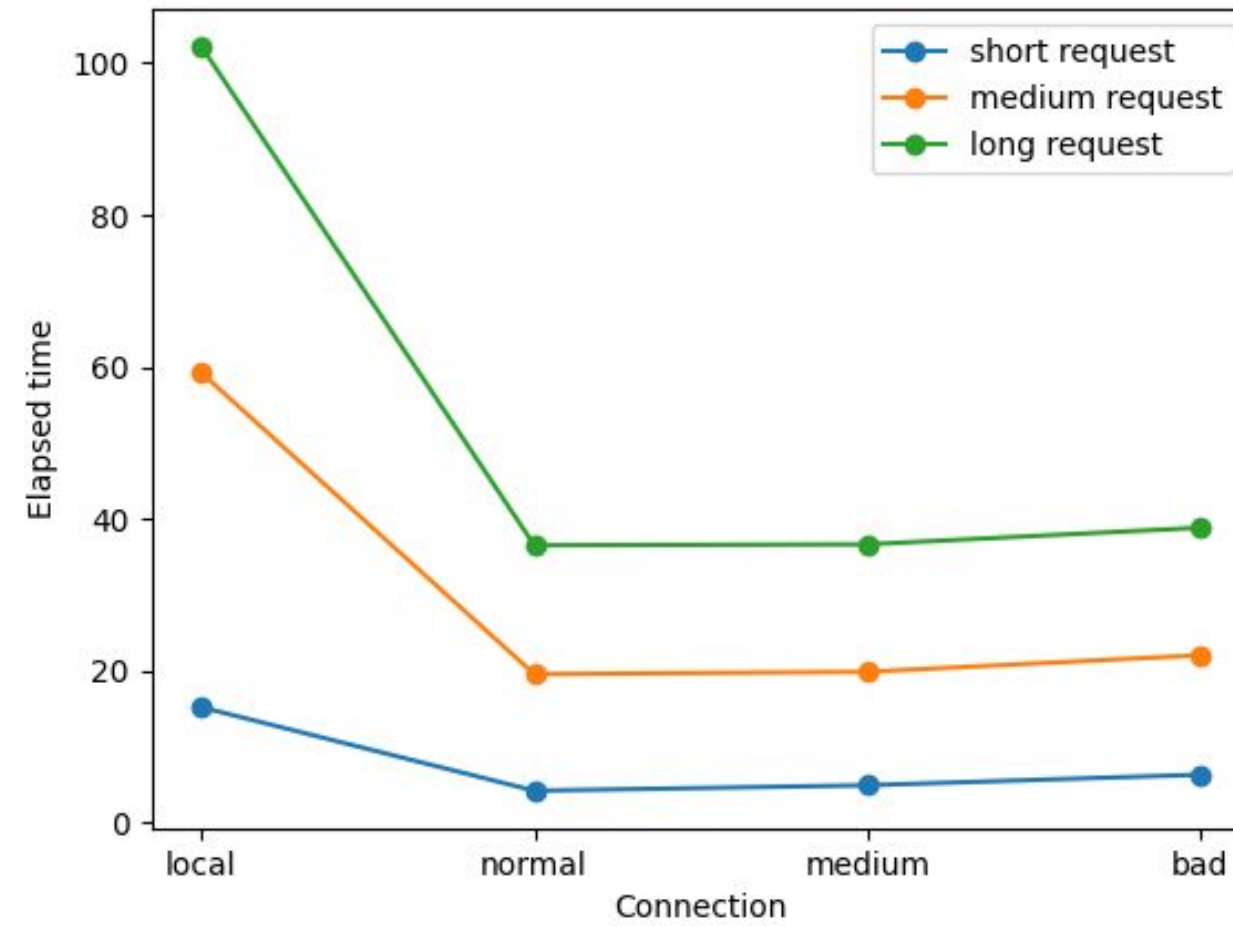


Considering various connection

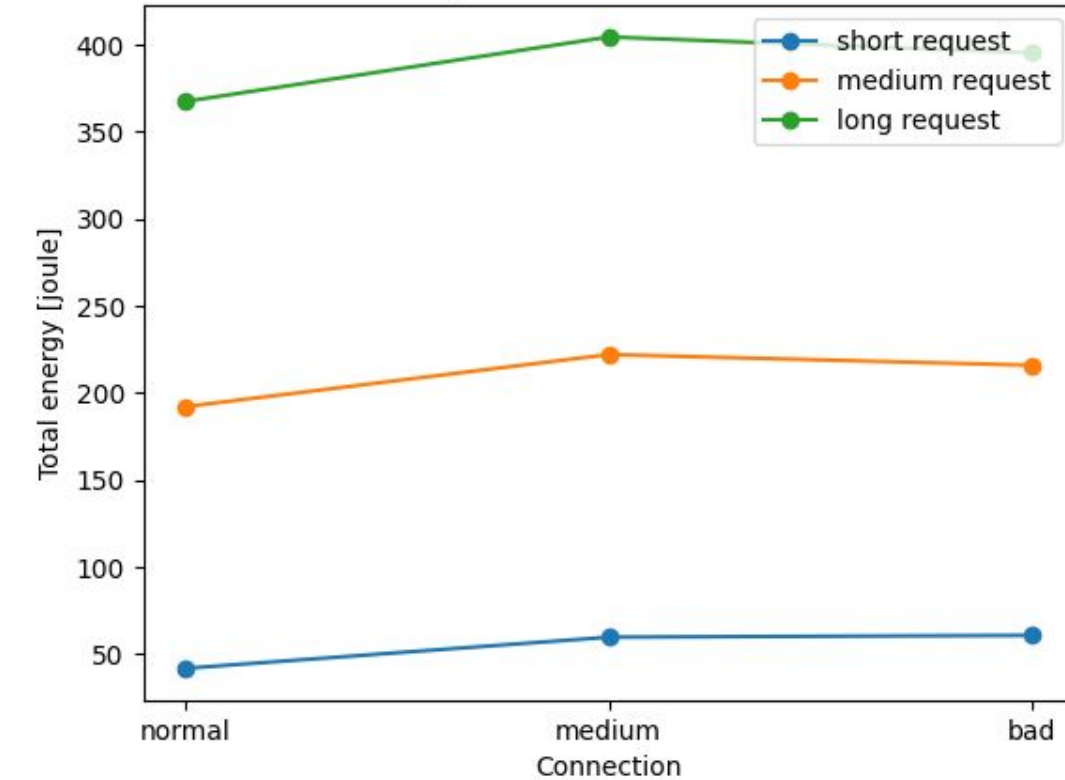
Total CPU Usage [%] result for client



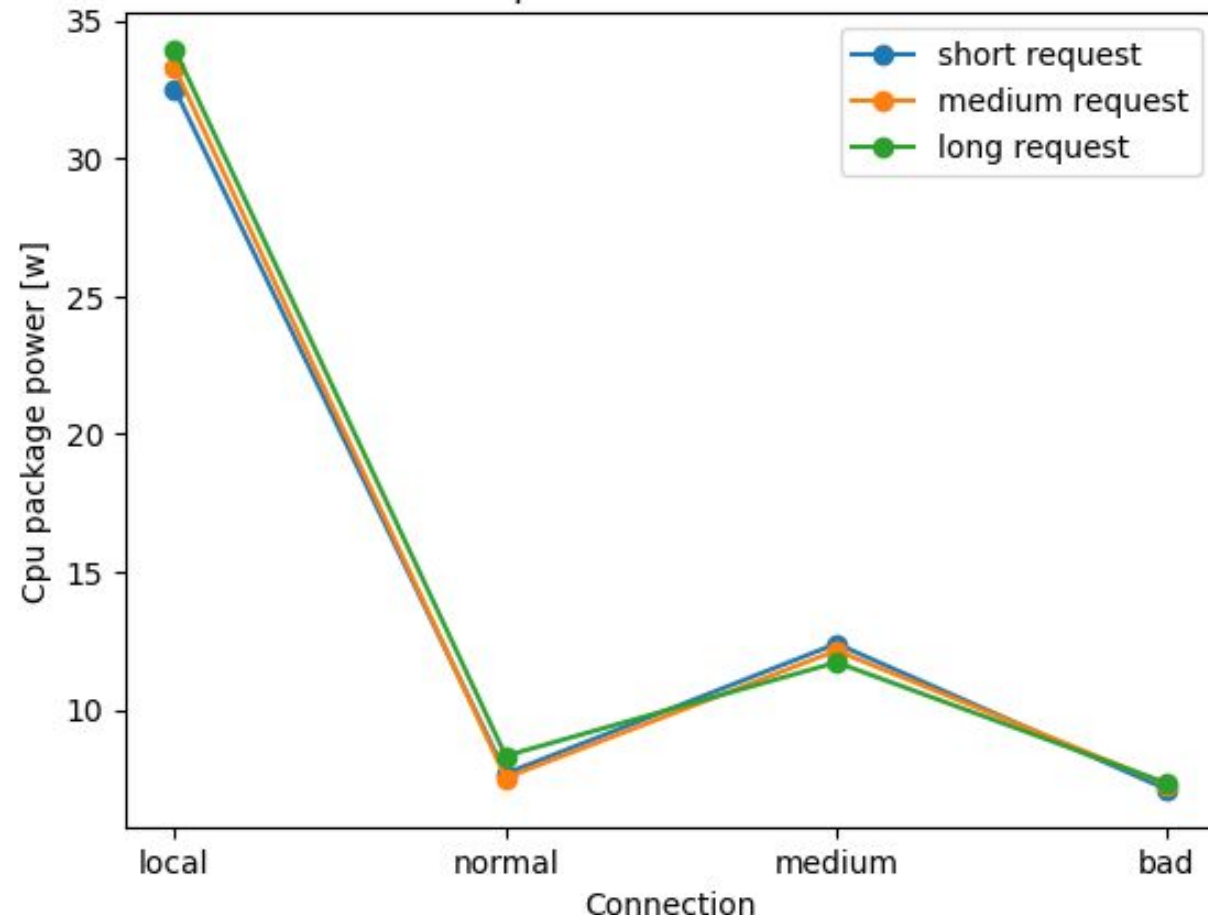
Time result for client



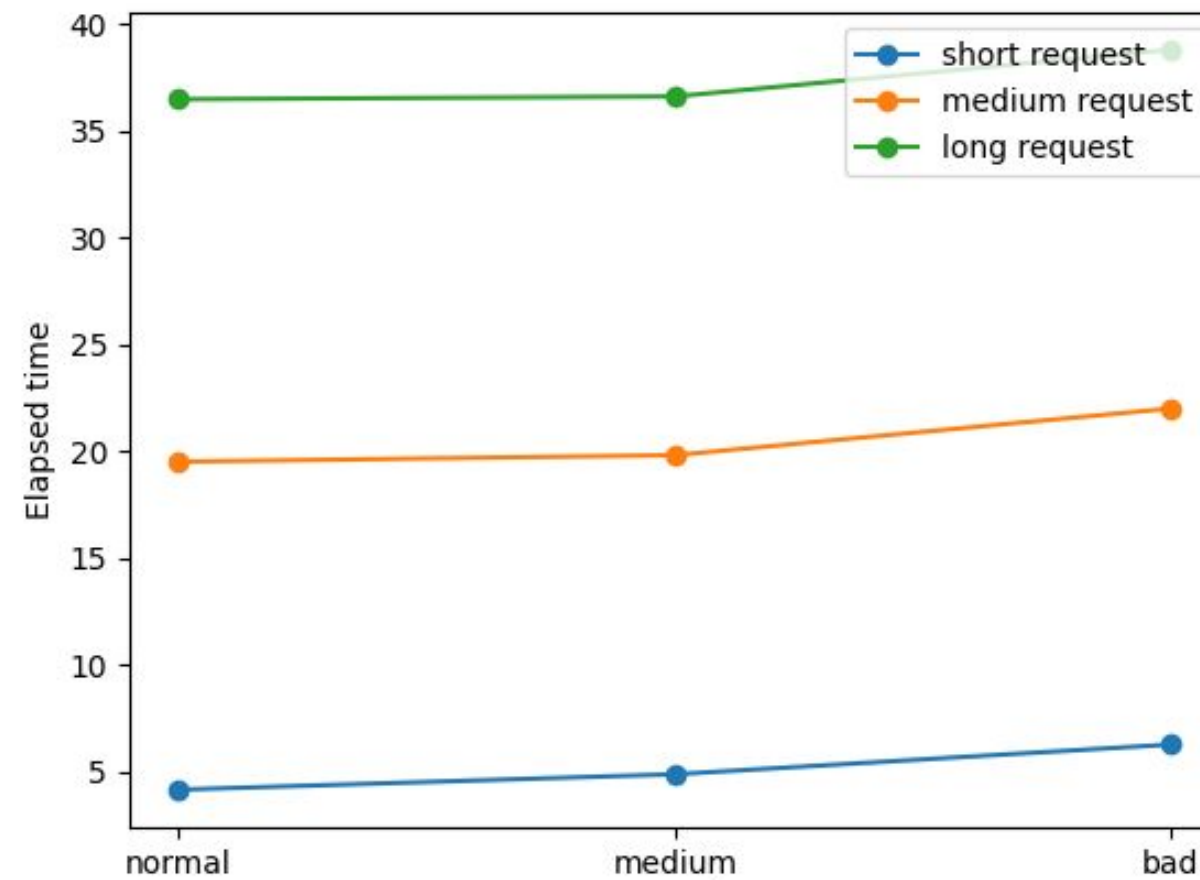
Energy remote result for client



CPU power result for client

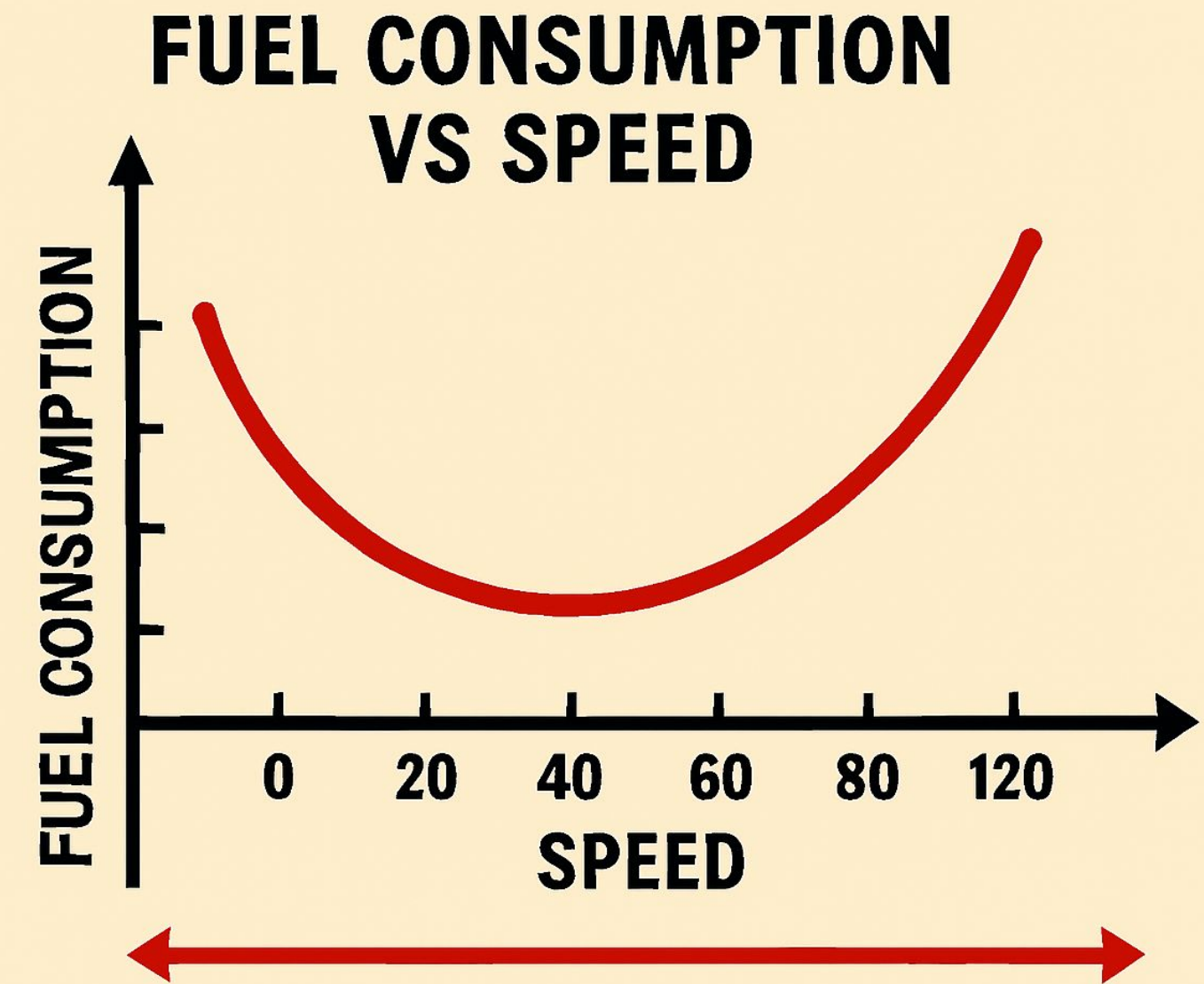
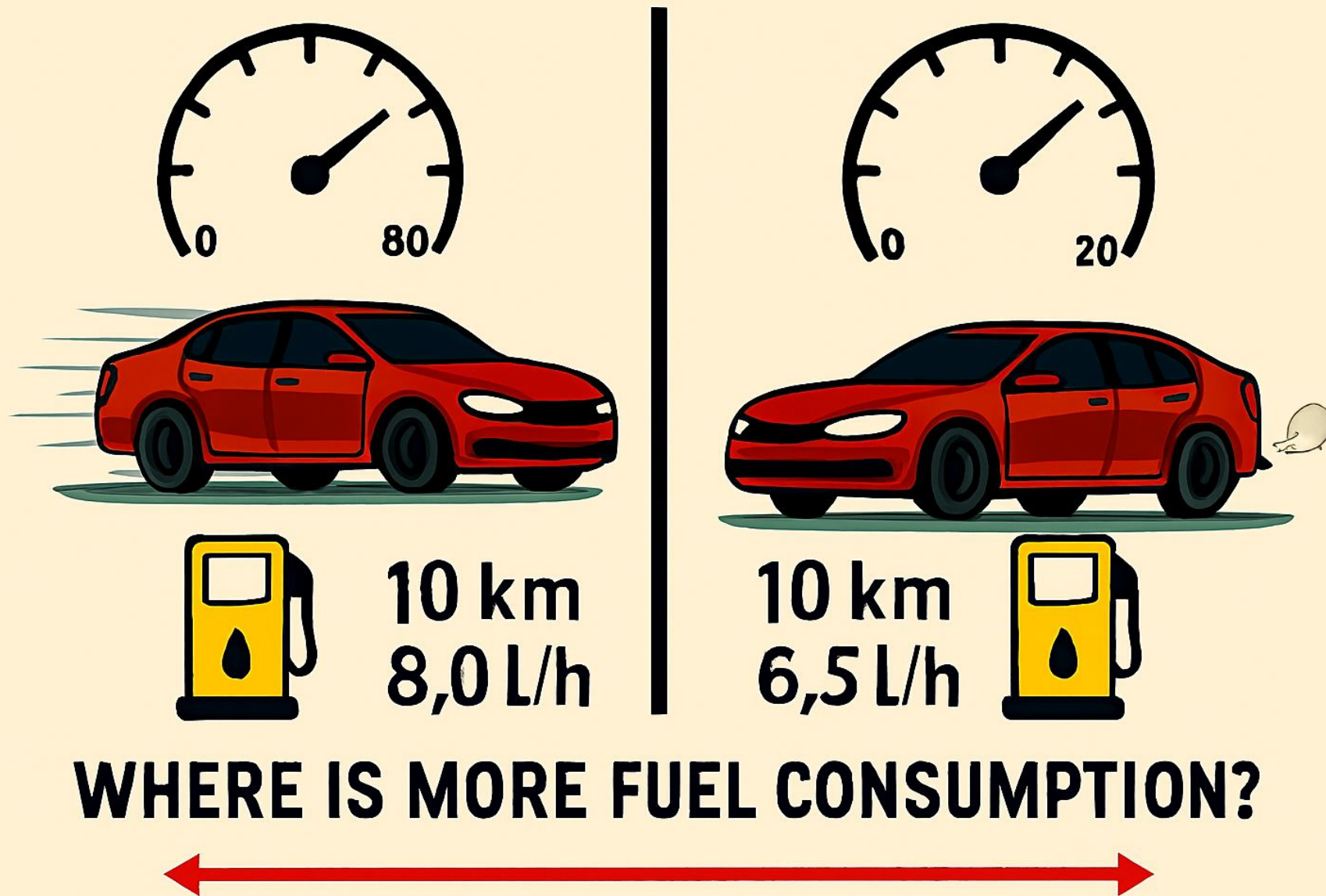


Time result for client



$$\text{Energy} = \text{Power} * \text{time}$$

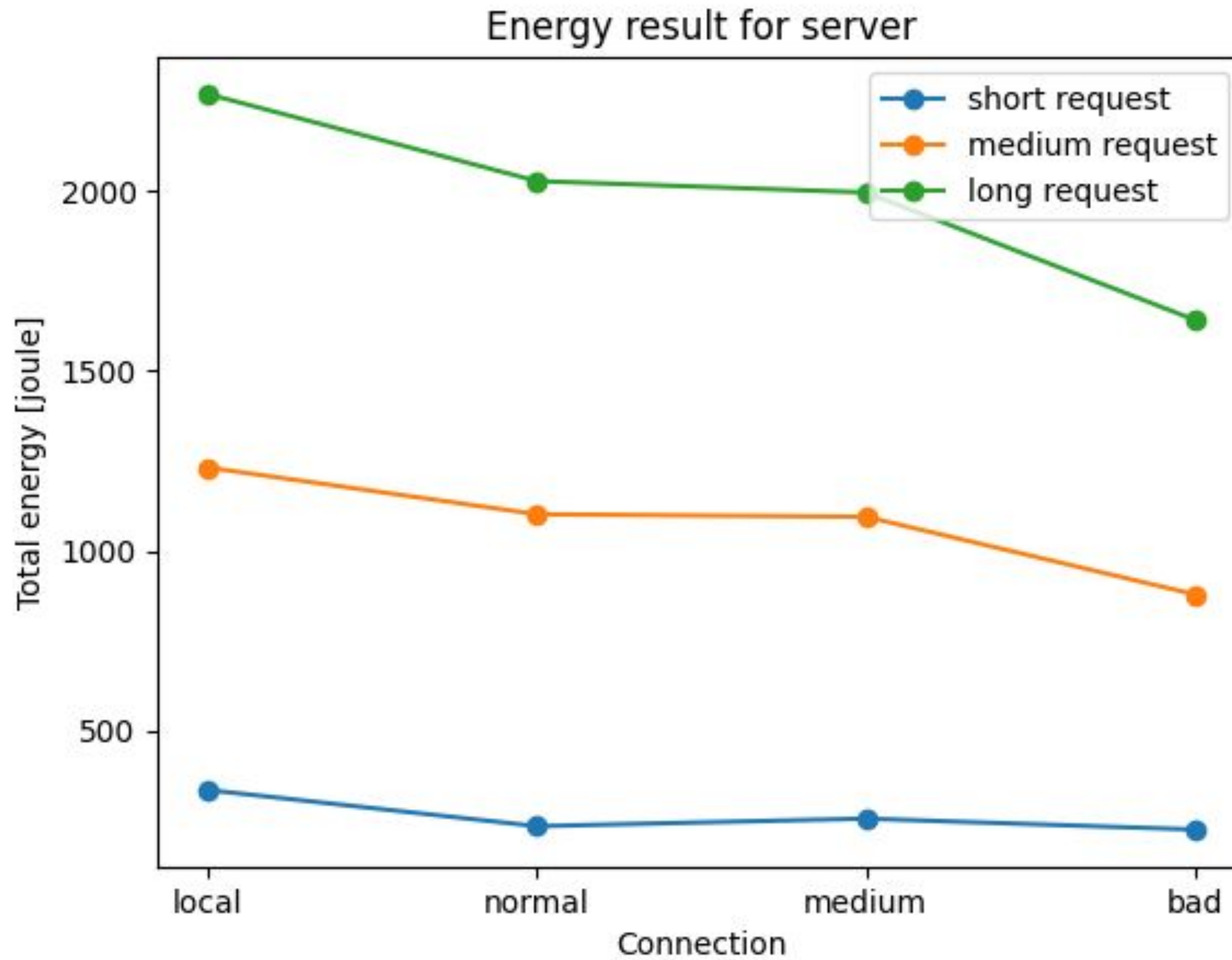
Observations



$$\text{Fuel_loss} = \text{consumption} * \text{time}$$

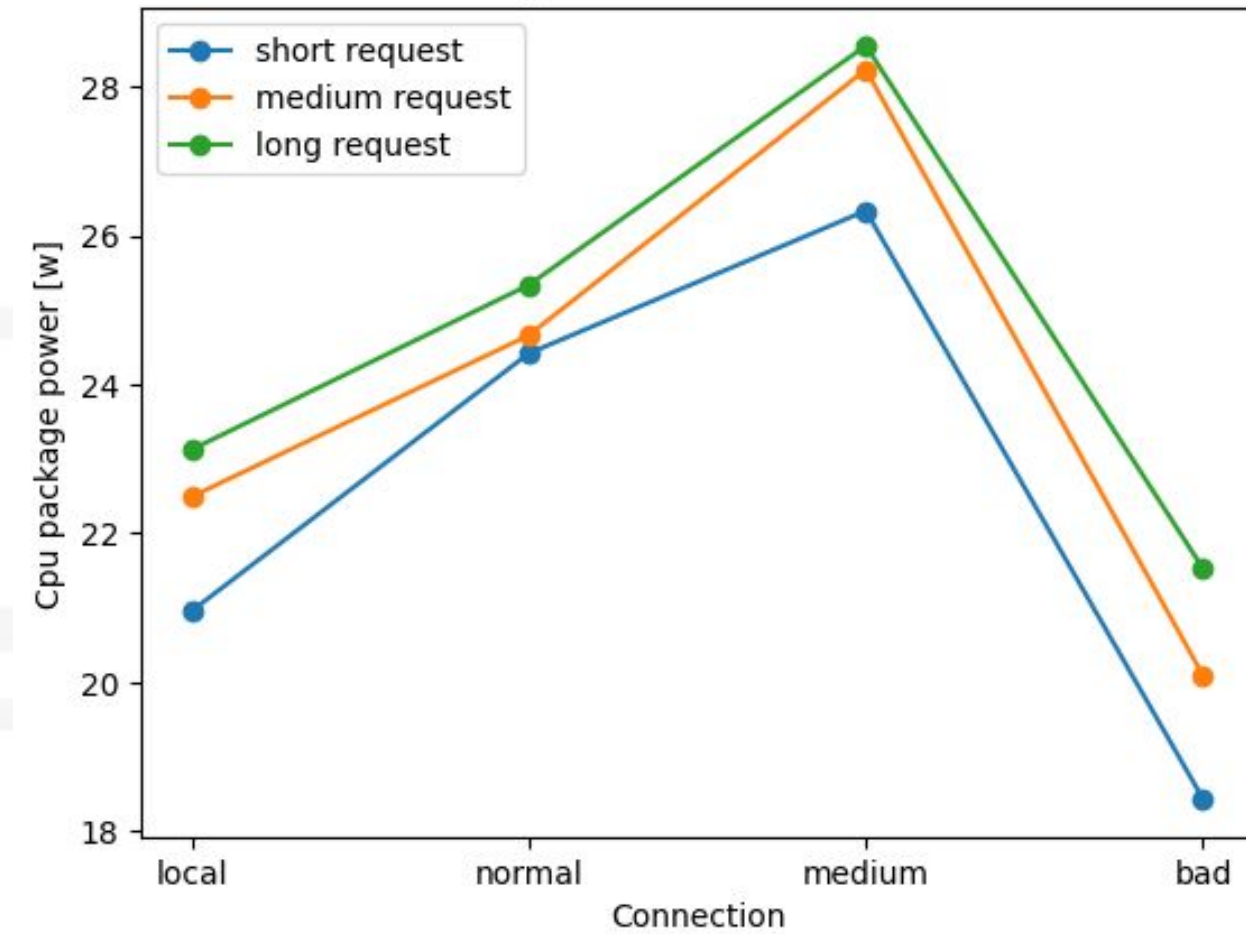
$$\text{Energy} = \text{Power} * \text{time}$$

Server

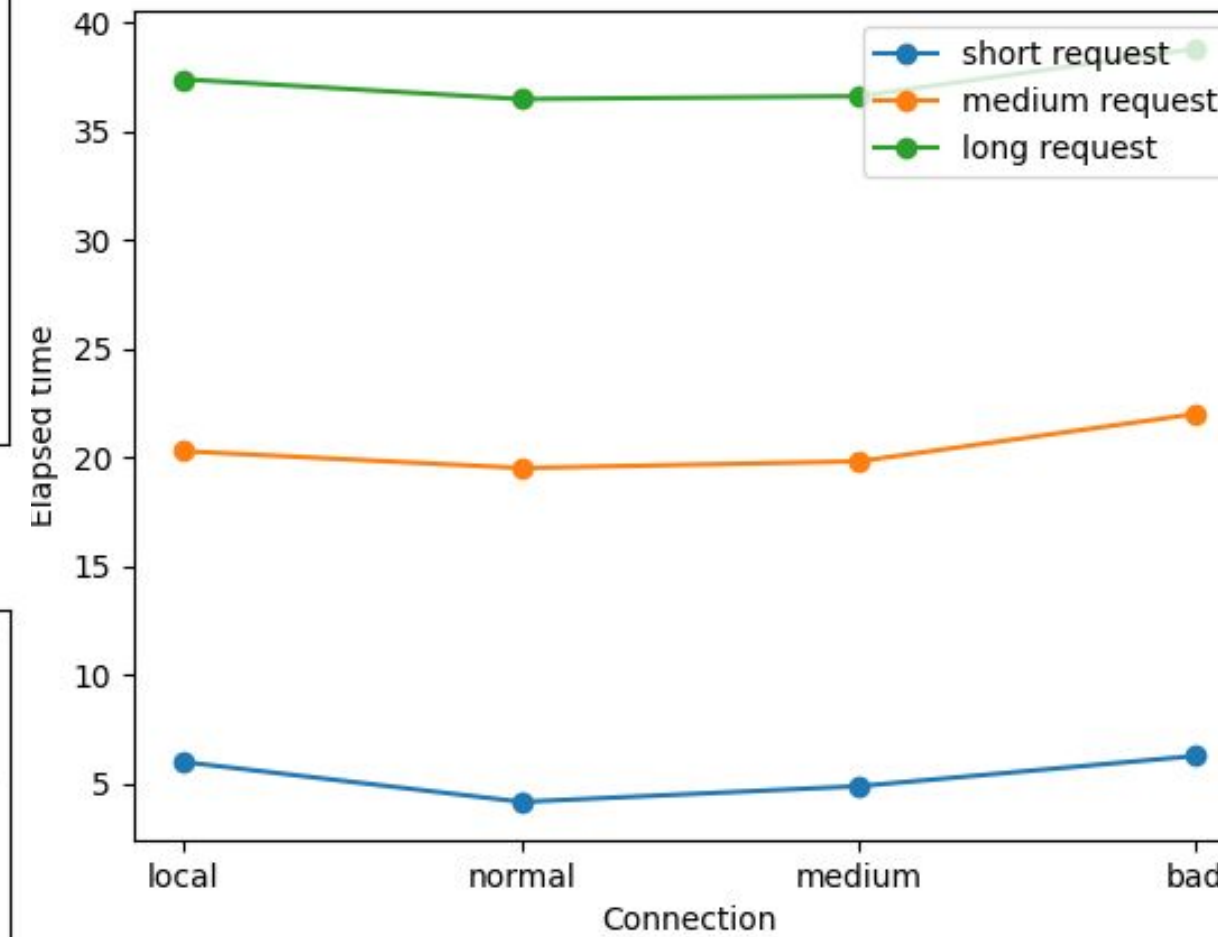


Server

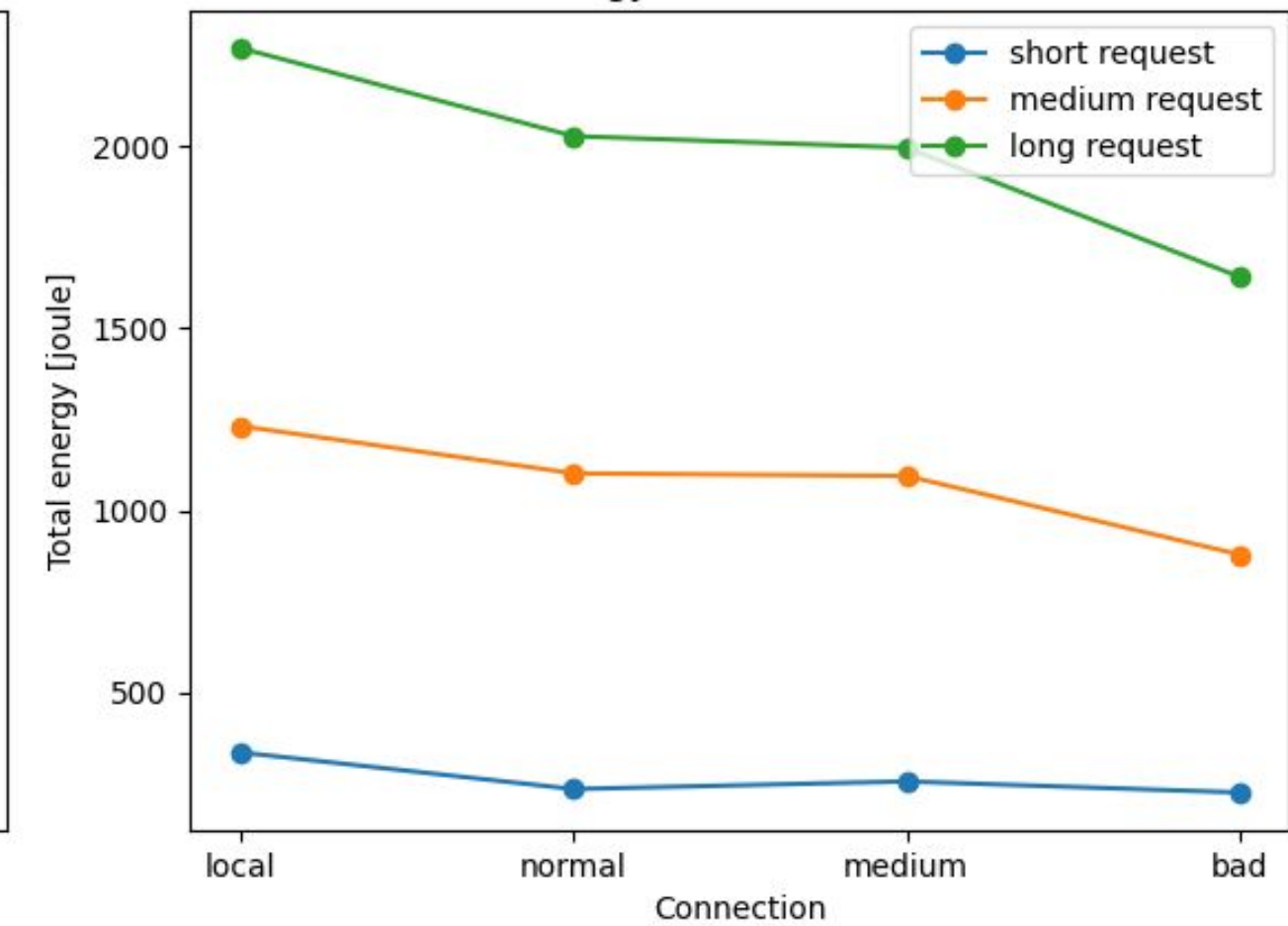
CPU power result for server



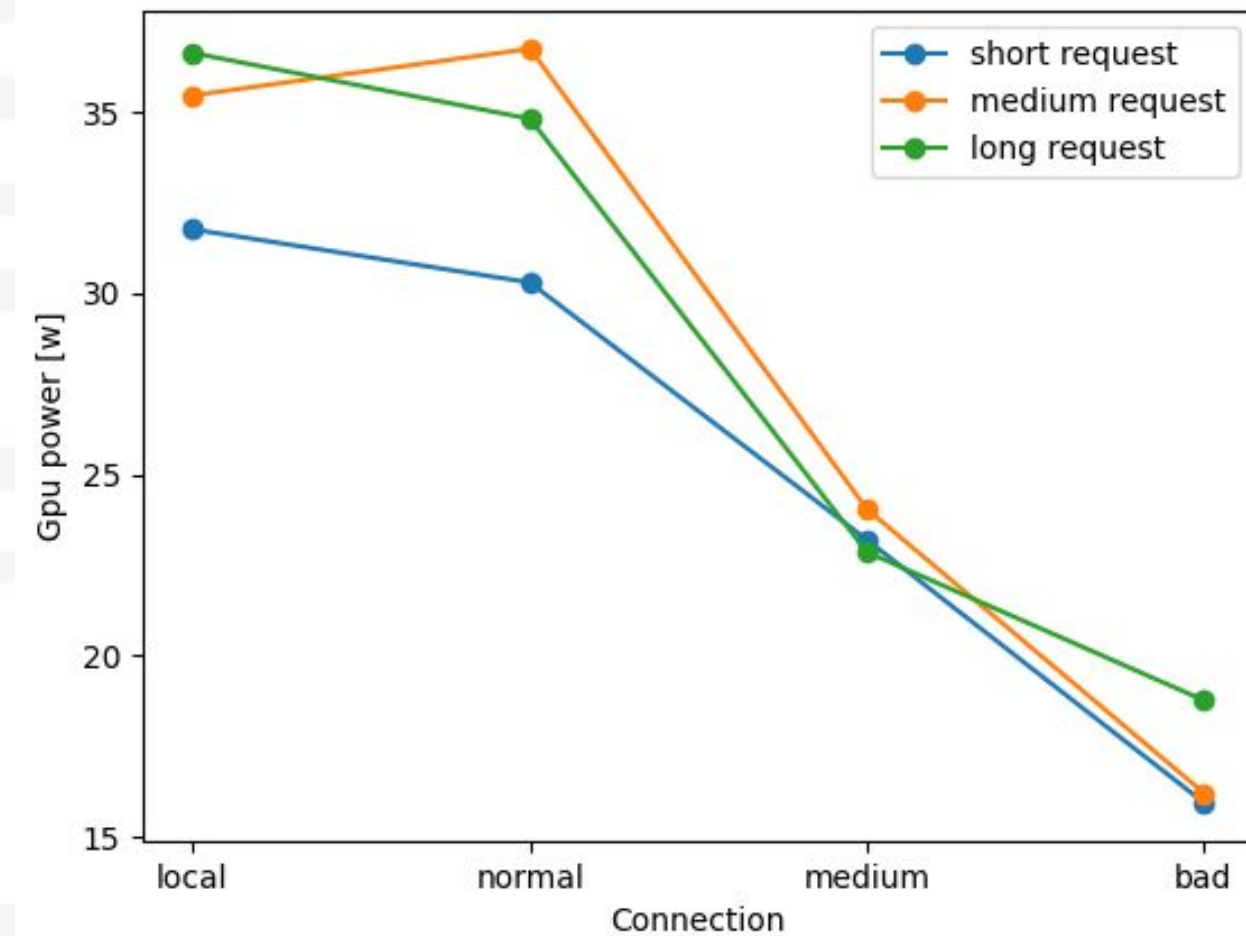
Time result for server



Energy result for server



GPU Power for server

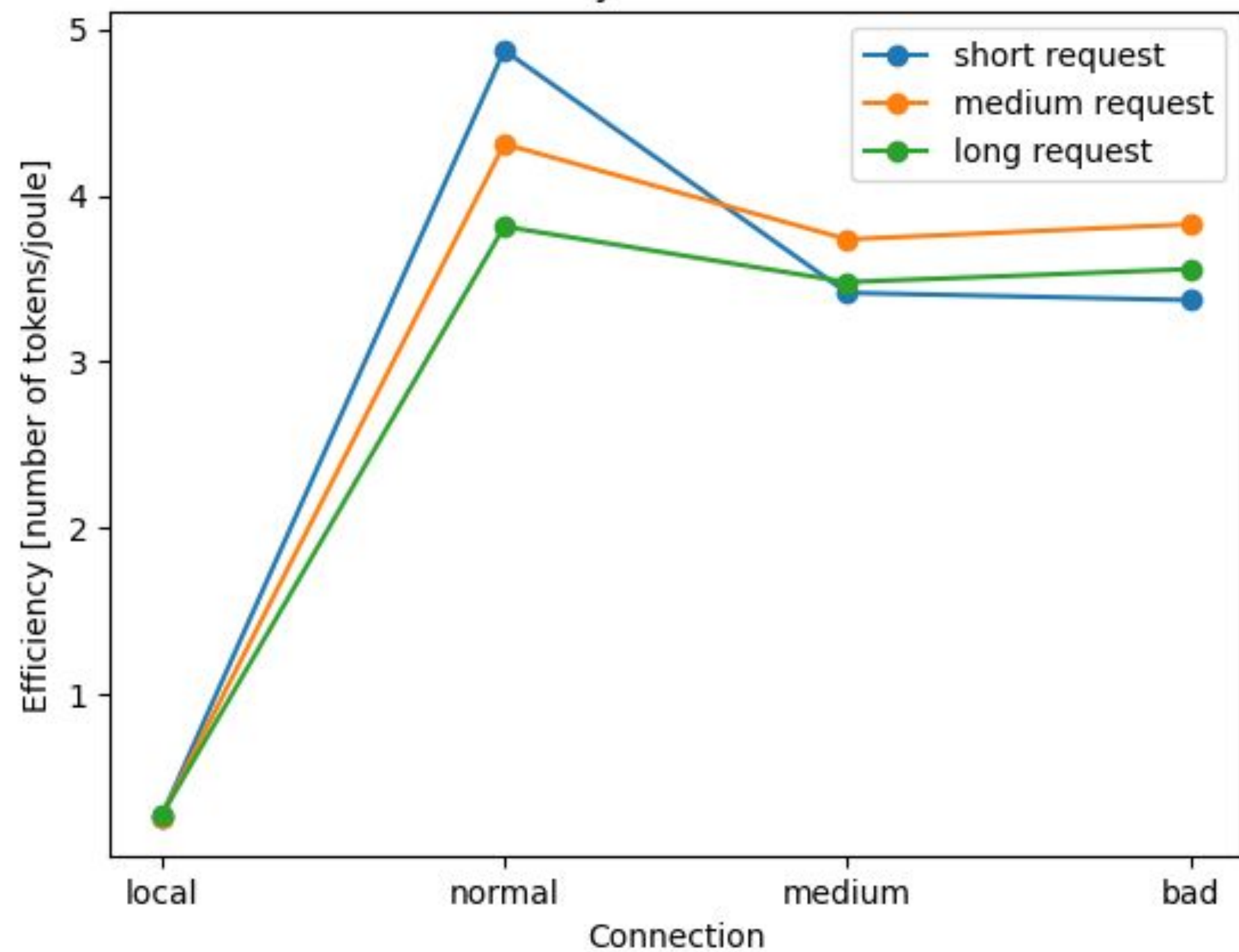


$$\text{Power} * \text{time} = \text{Energy}$$

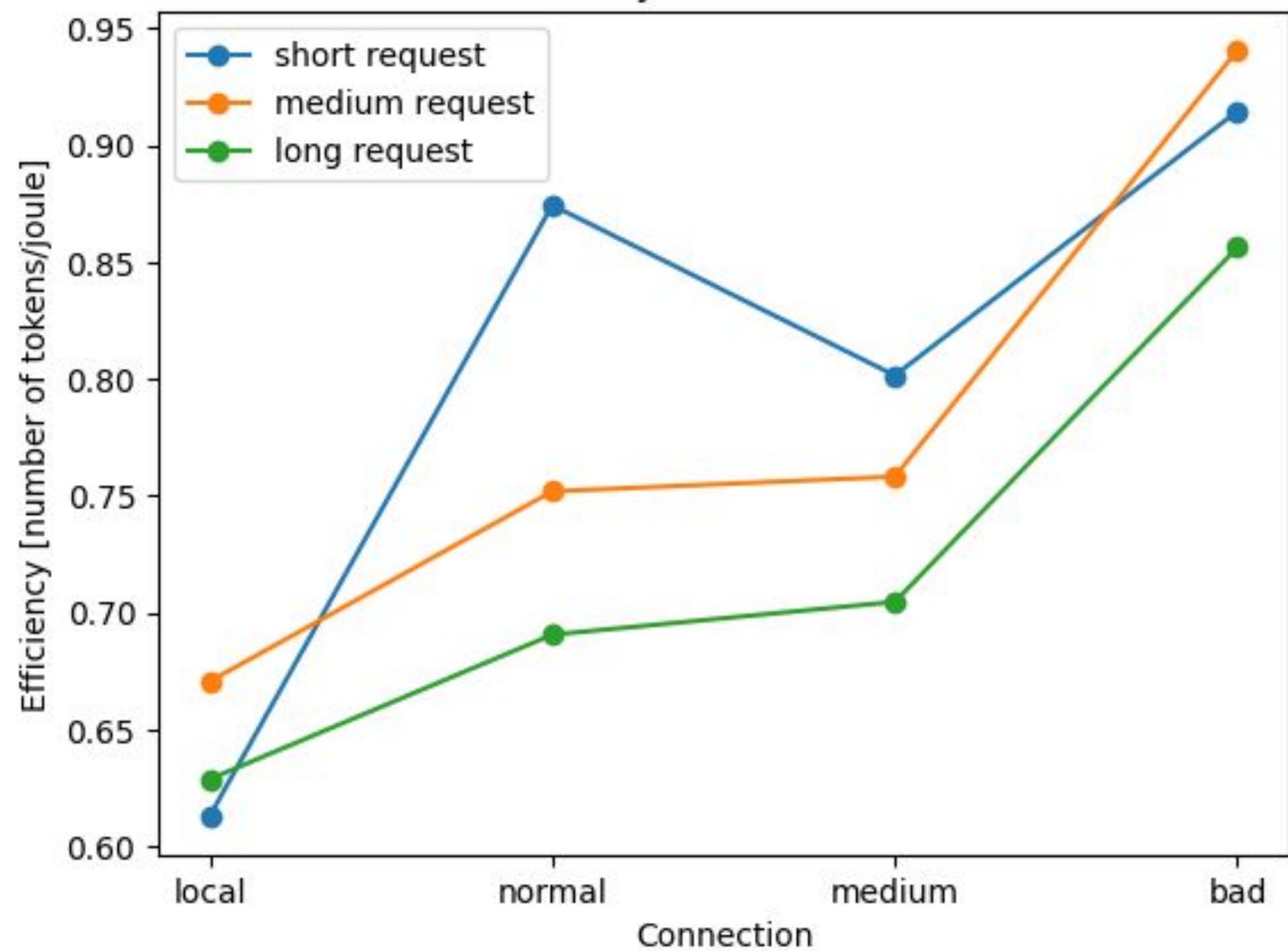


Efficiency

Efficiency result for client



Efficiency result for server



Conclusions

For maximum energy savings:

- Computing remotely, even with a poor connection
 - don't try to get the best out of connection
- Balance of execution time and power
- Use economic models
- Easy requests with short answers

Possible continuation of the research

- Consider more sources (disks, WiFi module, etc.) or the entire system
- Eternal energy measurement with a Wattmeter
- Consider a continuously operating system
- Find out which models consume more and what this depends on



Thank you

“Slow and steady wins the race”

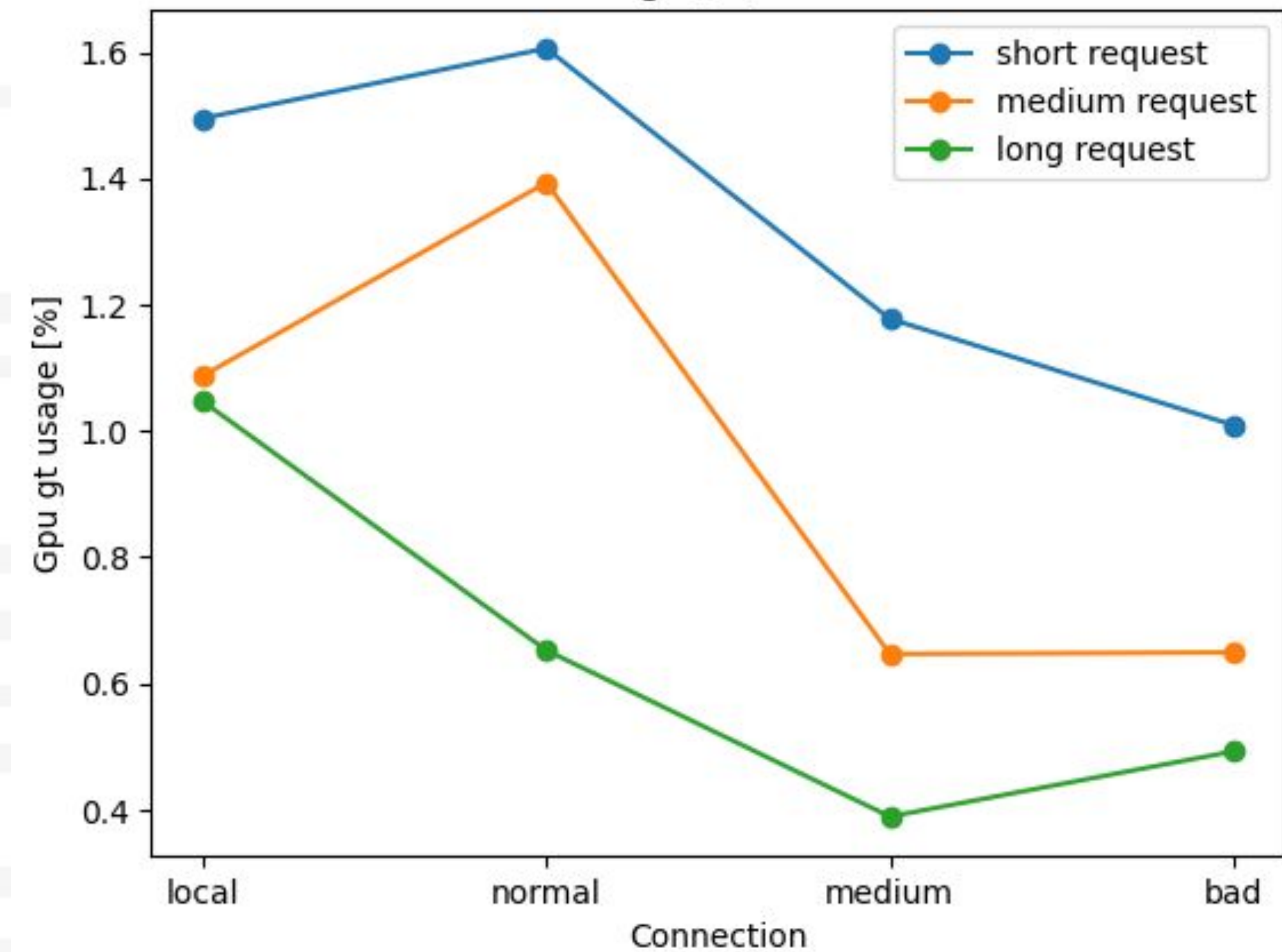
VS

“The sooner, the better”

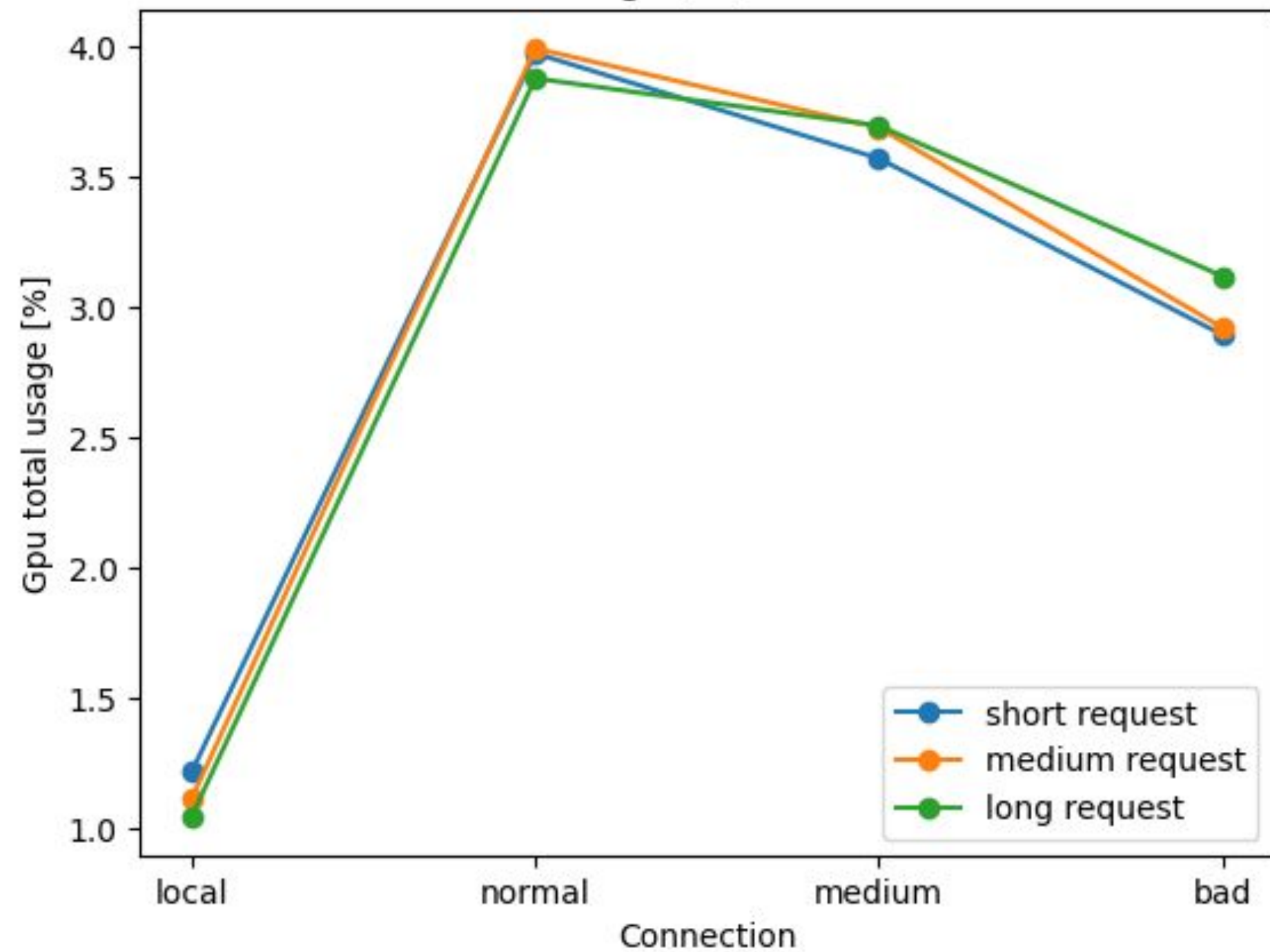
Backup slides

GPU Usage

Total GPU Usage [%] result for client

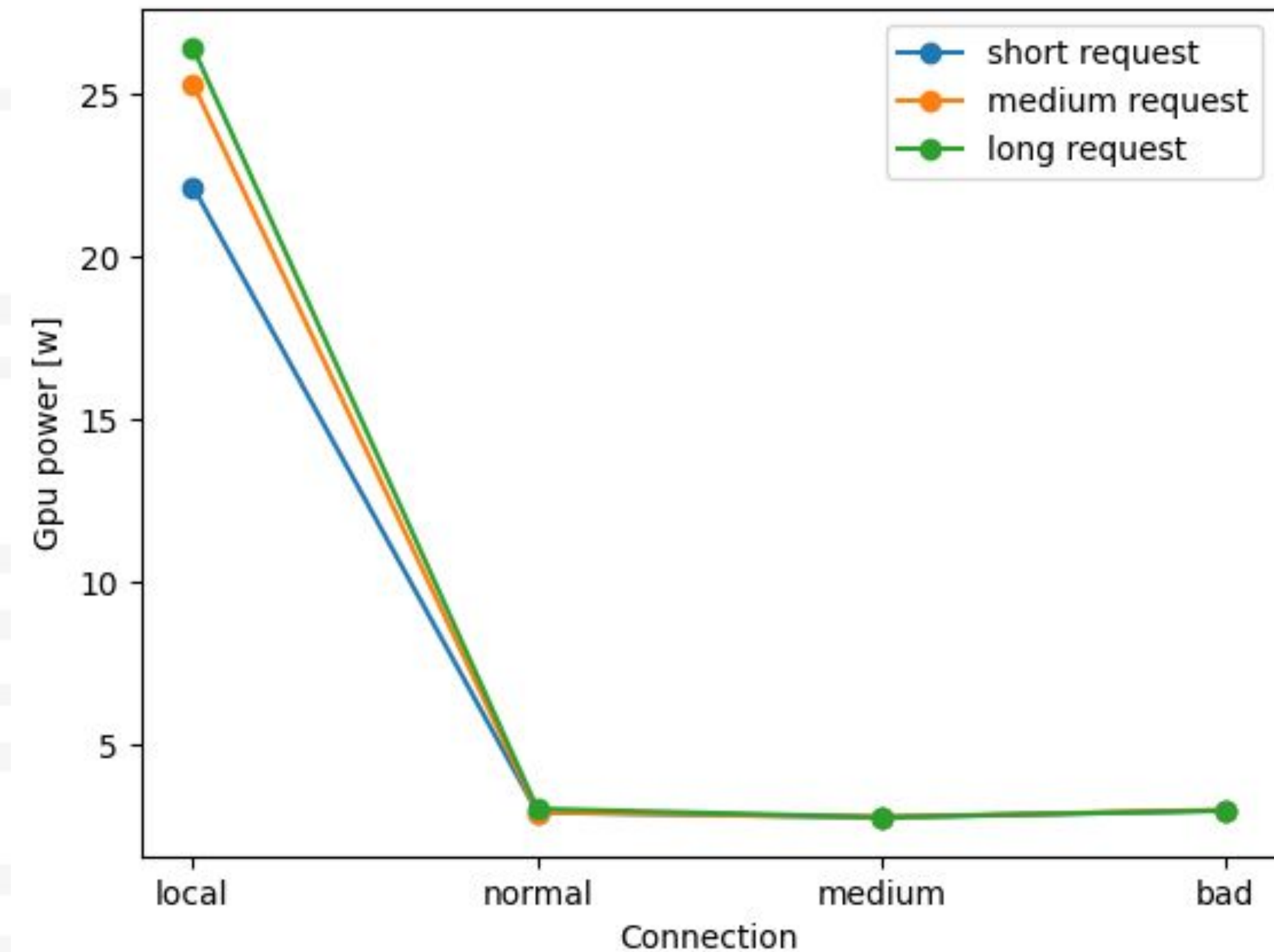


Total GPU Usage [%] result for server

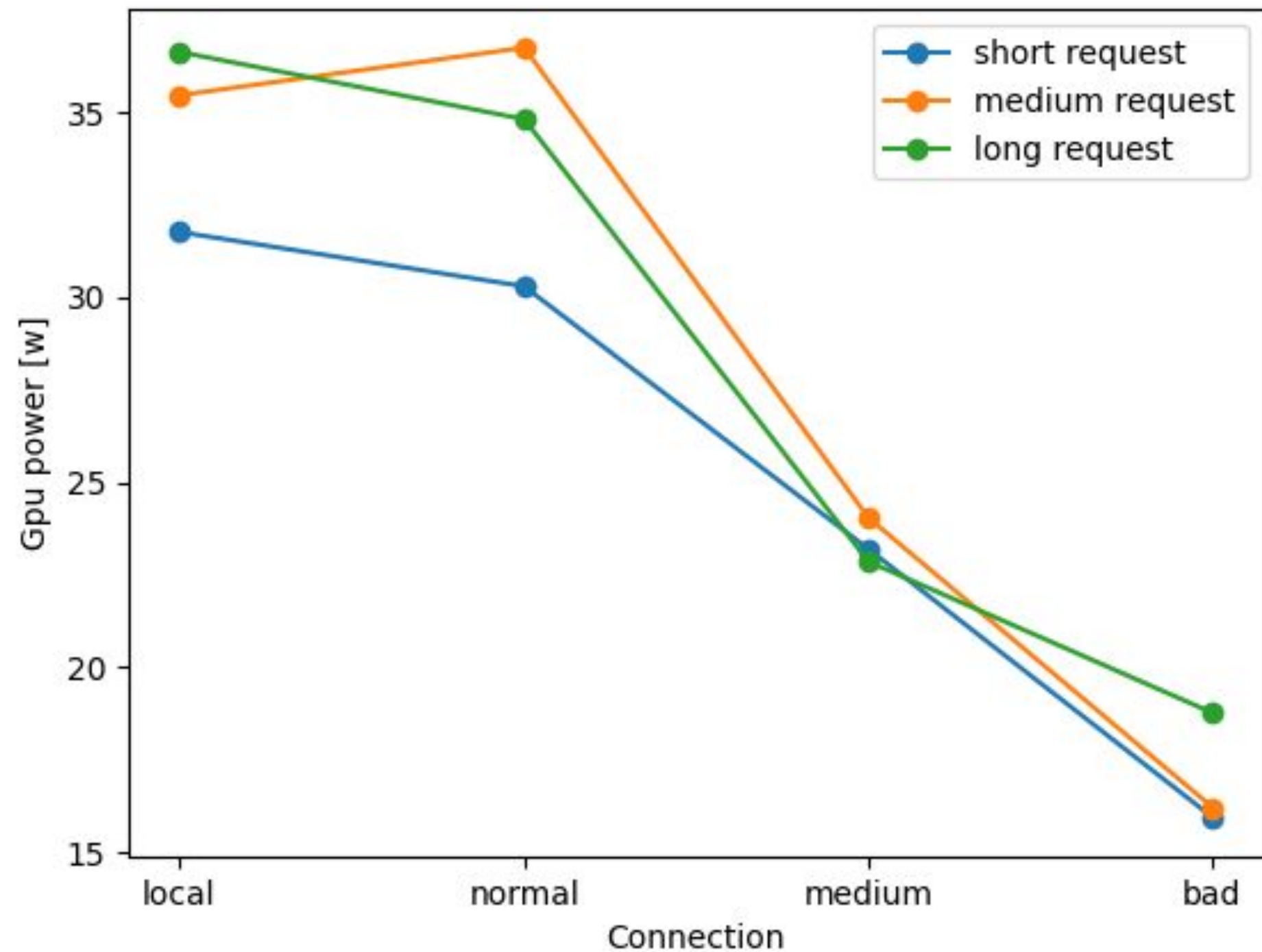


GPU Power

GPU Power for client

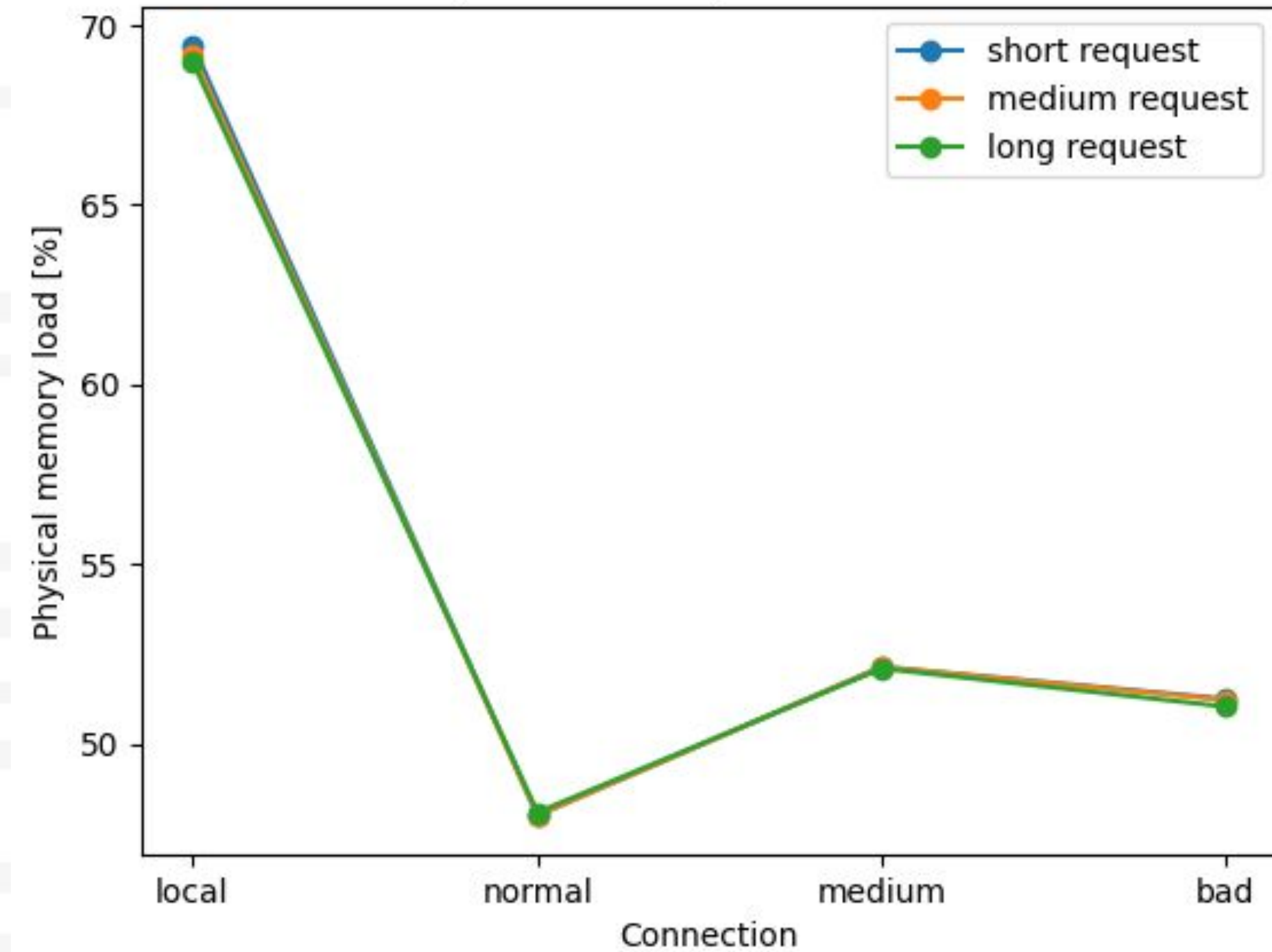


GPU Power for server

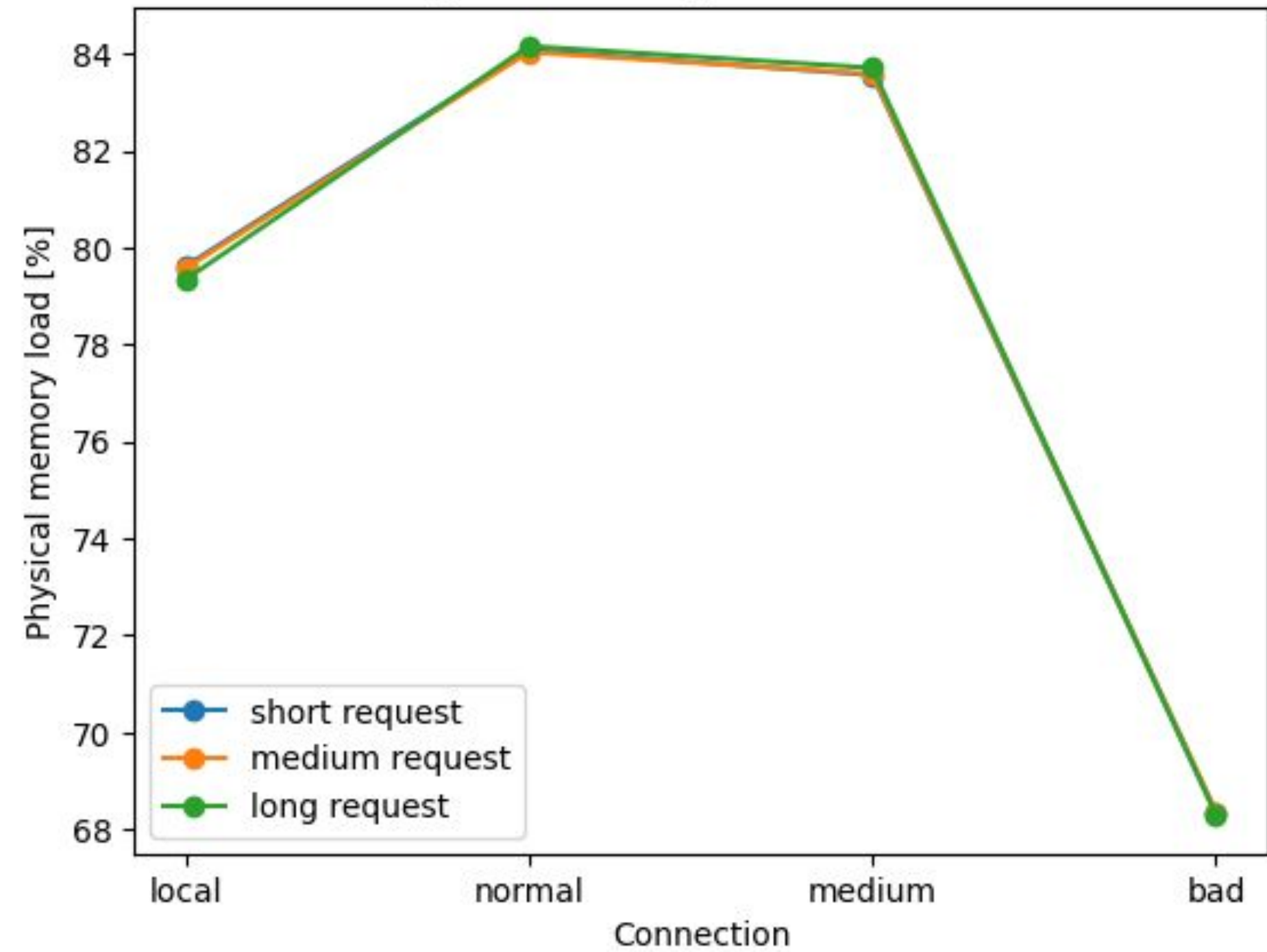


Physical Memory Load

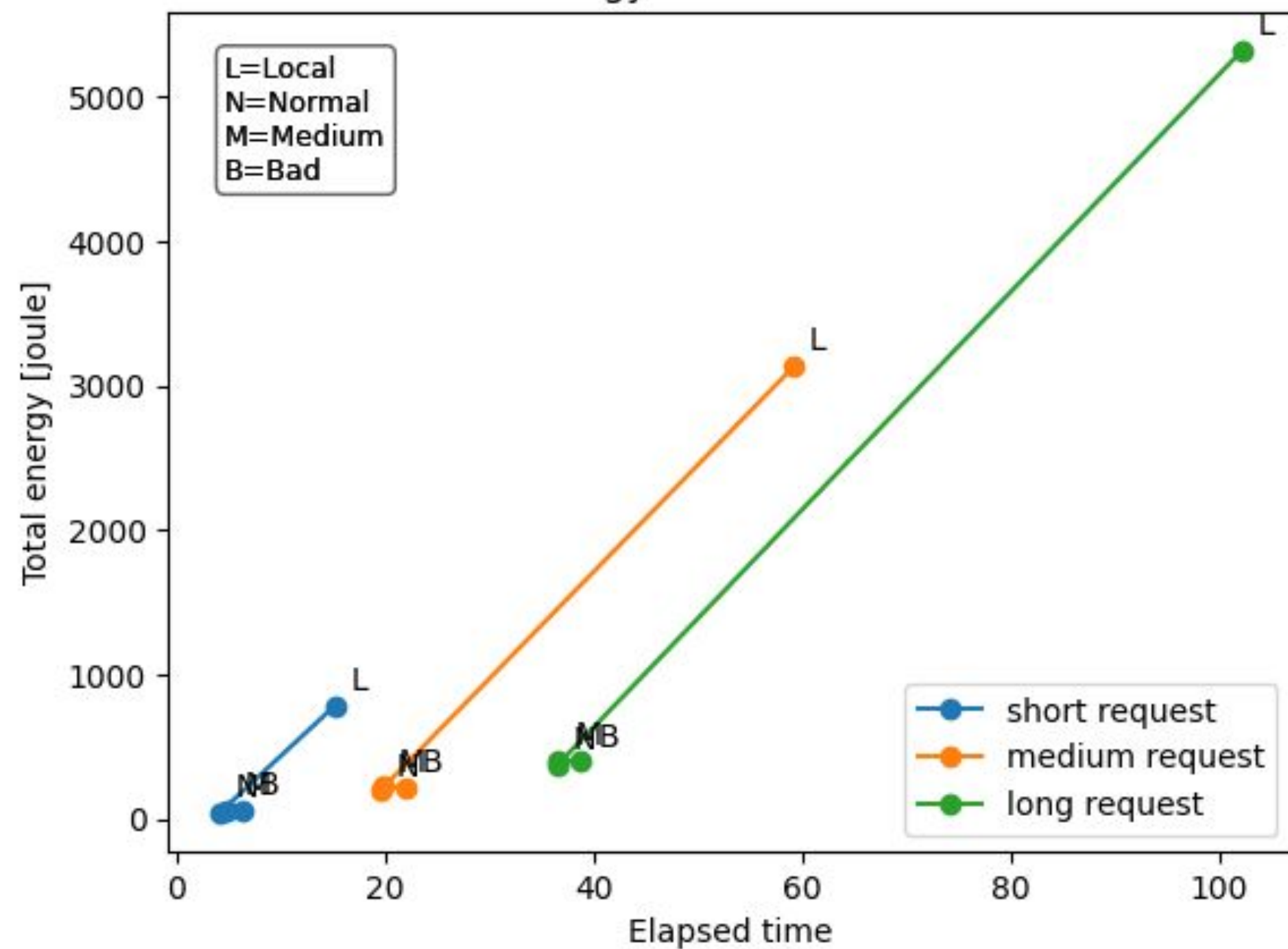
Physical Memory Load for client



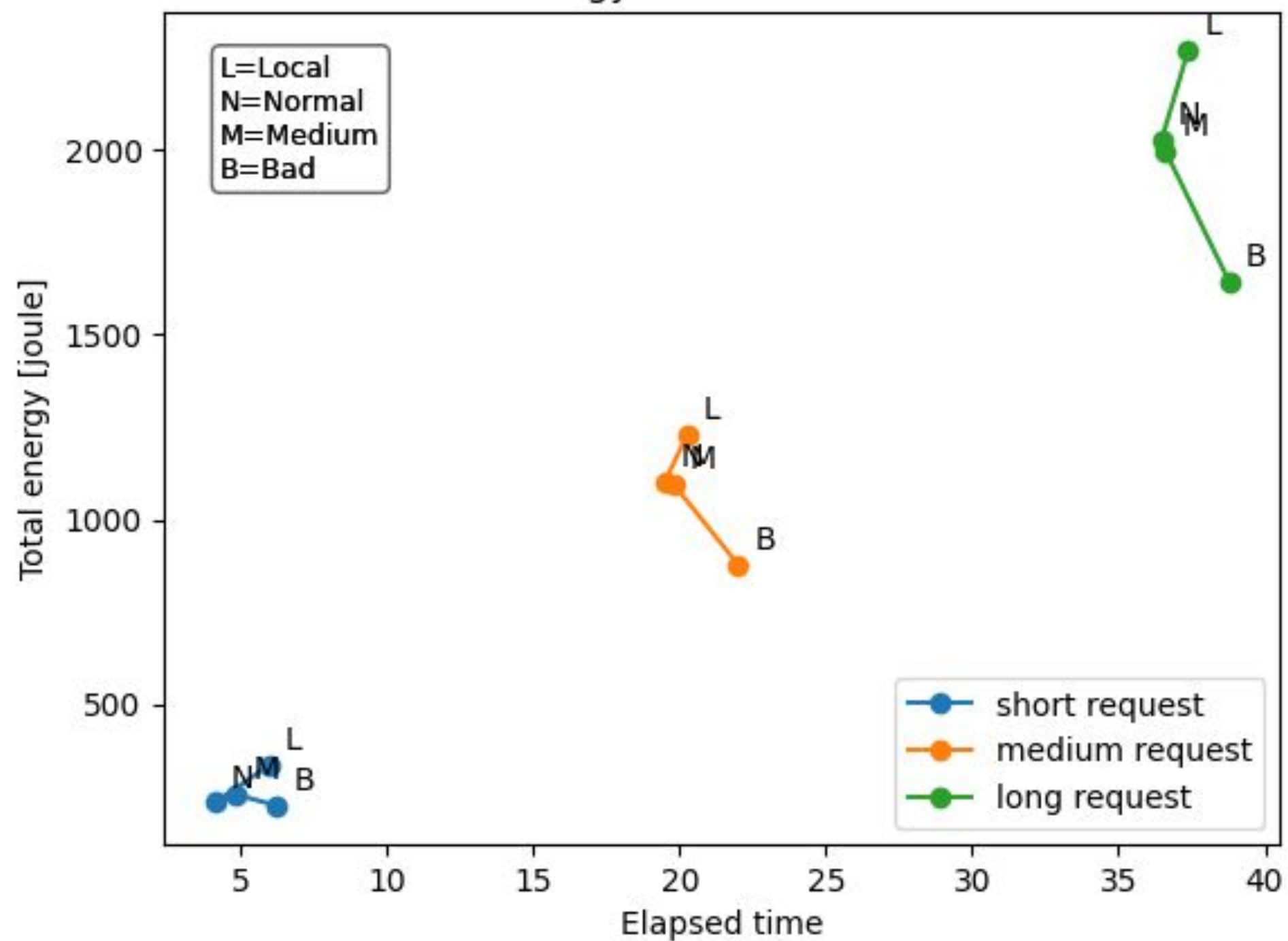
Physical Memory Load for server



Energy result for client



Energy result for server



Icone pronte all'uso

