


**www.phoronix-test-suite.com**

# ds lkey

AMD Ryzen 7 PRO 6850U testing with a LENOVO 21CM0001US (R22ET51W 1.21 BIOS) and AMD Radeon 680M 1GB on Ubuntu 22.10 via the Phoronix Test Suite.

**Automated Executive Summary**

*b had the most wins, coming in first place for 43% of the tests.*

*Based on the geometric mean of all complete results, the fastest (a) was 1.008x the speed of the slowest (c). b was 0.992x the speed of a and c was 1x the speed of b.*

*The results with the greatest spread from best to worst included:*

*KeyDB (Test: LPOP - Parallel Connections: 500) at 1.144x*
*KeyDB (Test: LPOP - Parallel Connections: 100) at 1.131x*
*KeyDB (Test: LPOP - Parallel Connections: 900) at 1.121x*
*KeyDB (Test: LPUSH - Parallel Connections: 100) at 1.096x*
*KeyDB (Test: LPOP - Parallel Connections: 50) at 1.085x*
*KeyDB (Test: LPUSH - Parallel Connections: 500) at 1.056x*
*KeyDB (Test: SET - Parallel Connections: 100) at 1.055x*
*KeyDB (Test: HMSET - Parallel Connections: 100) at 1.029x*
*KeyDB (Test: SADD - Parallel Connections: 100) at 1.016x*

*KeyDB (Test: SET - Parallel Connections: 500) at 1.016x.*

# Test Systems:

## a

## b

## c

Processor: AMD Ryzen 7 PRO 6850U @ 4.77GHz (8 Cores / 16 Threads), Motherboard: LENOVO 21CM0001US (R22ET51W 1.21 BIOS), Chipset: AMD Device 14b5, Memory: 16GB, Disk: 512GB Micron MTFDKBA512TFK, Graphics: AMD Radeon 680M 1GB (2200/400MHz), Audio: AMD Rembrandt Radeon HD Audio, Network: Qualcomm QCNFA765

OS: Ubuntu 22.10, Kernel: 6.1.0-060100rc2daily20221028-generic (x86_64), Desktop: GNOME Shell 43.0, Display Server: X Server + Wayland, OpenGL: 4.6 Mesa 22.2.1 (LLVM 15.0.2 DRM 3.49), Vulkan: 1.3.224, Compiler: GCC 12.2.0, File-System: ext4, Screen Resolution: 1920x1200

Kernel Notes: Transparent Huge Pages: madvise
Compiler Notes: --build=x86_64-linux-gnu --disable-vtable-verify --disable-werror --enable-cet --enable-checking=release --enable-clocale=gnu --enable-default-pie --enable-gnu-unique-object --enable-languages=c,ada,c++,go,d,fortran,objc,obj-c++,m2 --enable-libphobos-checking=release --enable-libstdcxx-debug --enable-libstdcxx-time=yes --enable-multiarch --enable-multilib --enable-nls --enable-objc-gc=auto --enable-offload-defaulted --enable-offload-targets=nvptx-none=/build/gcc-12-U8K4Qv/gcc-12-12.2.0/debian/tmp-nvptx/usr,amdgcn-amdhsa=/build/gcc-12-U8K4Qv/gcc-12-12.2.0/debian/tmp-gcn/usr --enable-plugin --enable-shared --enable-threads=posix --host=x86_64-linux-gnu --program-prefix=x86_64-linux-gnu- --target=x86_64-linux-gnu --with-abi=m64 --with-arch-32=i686 --with-default-libstdcxx-abi=new --with-gcc-major-version-only --with-multilib-list=m32,m64,mx32 --with-target-system-zlib=auto --with-tune=generic --without-cuda-driver -v
Processor Notes: Scaling Governor: amd-pstate schedutil (Boost: Enabled) - Platform Profile: performance - CPU Microcode: 0xa404102 - ACPI Profile: performance
Python Notes: Python 3.10.7
Security Notes: itlb_multihit: Not affected + l1tf: Not affected + mds: Not affected + meltdown: Not affected + mmio_stale_data: Not affected + retbleed: Not affected + spec_store_bypass: Mitigation of SSB disabled via prctl + spectre_v1: Mitigation of usercopy/swapgs barriers and __user pointer sanitization + spectre_v2: Mitigation of Retpolines IBPB: conditional IBRS_FW STIBP: always-on RSB filling PBRSB-eIBRS: Not affected + srbds: Not affected + tsx_async_abort: Not affected

| | a | b | c |
|---|---|---|---|
| **KeyDB - HMSET - 900 (Reqs/sec)** | **201399** | **203707** | 202206 |
| **Normalized** | 98.87% | 100% | 99.26% |
| **KeyDB - HMSET - 500 (Reqs/sec)** | **205130** | 204631 | **203269** |
| **Normalized** | 100% | 99.76% | 99.09% |
| **KeyDB - HMSET - 100 (Reqs/sec)** | **220860** | **214635** | 219592 |
| **Normalized** | 100% | 97.18% | 99.43% |
| **KeyDB - LPUSH - 900 (Reqs/sec)** | **270349** | 269025 | **267054** |
| **Normalized** | 100% | 99.51% | 98.78% |
| **KeyDB - LPUSH - 500 (Reqs/sec)** | 277528 | **285523** | **270481** |
| **Normalized** | 97.2% | 100% | 94.73% |
| **KeyDB - LPOP - 500 (Reqs/sec)** | **319444** | **279348** | 281265 |
| **Normalized** | 100% | 87.45% | 88.05% |

# ds lkey

| | | | |
|---|---|---|---|
| **KeyDB - LPOP - 900 (Reqs/sec)** | **313710** | **279925** | 289251 |
| **Normalized** | 100% | 89.23% | 92.2% |
| **KeyDB - SET - 900 (Reqs/sec)** | **297451** | **294012** | 294822 |
| **Normalized** | 100% | 98.84% | 99.12% |
| **KeyDB - SET - 500 (Reqs/sec)** | **302519** | **297642** | 300517 |
| **Normalized** | 100% | 98.39% | 99.34% |
| **KeyDB - HMSET - 50 (Reqs/sec)** | 301397 | **301688** | **299625** |
| **Normalized** | 99.9% | 100% | 99.32% |
| **KeyDB - LPUSH - 100 (Reqs/sec)** | **318624** | **290676** | 299638 |
| **Normalized** | 100% | 91.23% | 94.04% |
| **KeyDB - SADD - 900 (Reqs/sec)** | **307443** | **304371** | 305295 |
| **Normalized** | 100% | 99% | 99.3% |
| **KeyDB - GET - 900 (Reqs/sec)** | **308277** | **305945** | 306020 |
| **Normalized** | 100% | 99.24% | 99.27% |
| **KeyDB - LPOP - 100 (Reqs/sec)** | **326757** | **288964** | 307918 |
| **Normalized** | 100% | 88.43% | 94.23% |
| **KeyDB - SADD - 500 (Reqs/sec)** | **309906** | **311701** | 310928 |
| **Normalized** | 99.42% | 100% | 99.75% |
| **KeyDB - GET - 500 (Reqs/sec)** | 312161 | **310492** | **312735** |
| **Normalized** | 99.82% | 99.28% | 100% |
| **KeyDB - SET - 100 (Reqs/sec)** | 316036 | **323219** | **306421** |
| **Normalized** | 97.78% | 100% | 94.8% |
| **KeyDB - SADD - 100 (Reqs/sec)** | **314193** | **319358** | 314828 |
| **Normalized** | 98.38% | 100% | 98.58% |
| **KeyDB - GET - 100 (Reqs/sec)** | 317398 | **318068** | **315932** |
| **Normalized** | 99.79% | 100% | 99.33% |
| **KeyDB - LPUSH - 50 (Reqs/sec)** | **377803** | **378527** | 378092 |
| **Normalized** | 99.81% | 100% | 99.89% |
| **KeyDB - LPOP - 50 (Reqs/sec)** | **399629** | 370255 | **368172** |
| **Normalized** | 100% | 92.65% | 92.13% |
| **KeyDB - SET - 50 (Reqs/sec)** | **391239** | 389463 | **389290** |
| **Normalized** | 100% | 99.55% | 99.5% |
| **KeyDB - SADD - 50 (Reqs/sec)** | **395707** | 395848 | **396228** |
| **Normalized** | 99.87% | 99.9% | 100% |
| **KeyDB - GET - 50 (Reqs/sec)** | **400375** | 397769 | **396253** |
| **Normalized** | 100% | 99.35% | 98.97% |
| **Neural Magic DeepSparse - N.Q.A.B.b.u.S.1.P - A.M.S (ms/batch)** | **199.16** | 197.8215 | **197.7918** |
| **Normalized** | 99.31% | 99.98% | 100% |
| **Neural Magic DeepSparse - N.Q.A.B.b.u.S.1.P - A.M.S (items/sec)** | **20.0433** | 20.1989 | **20.2206** |
| **Normalized** | 99.12% | 99.89% | 100% |
| **Neural Magic DeepSparse - N.S.A.8.P.Q.B.B.U - A.M.S (ms/batch)** | **61.5757** | 61.6214 | **61.9249** |
| **Normalized** | 100% | 99.93% | 99.44% |
| **Neural Magic DeepSparse - N.S.A.8.P.Q.B.B.U - A.M.S (items/sec)** | **64.9385** | 64.8899 | **64.5712** |
| **Normalized** | 100% | 99.93% | 99.43% |
| **Neural Magic DeepSparse - N.D.C.o.b.u.o.I - A.M.S (ms/batch)** | 774.2867 | **773.5511** | **775.2753** |
| **Normalized** | 99.9% | 100% | 99.78% |

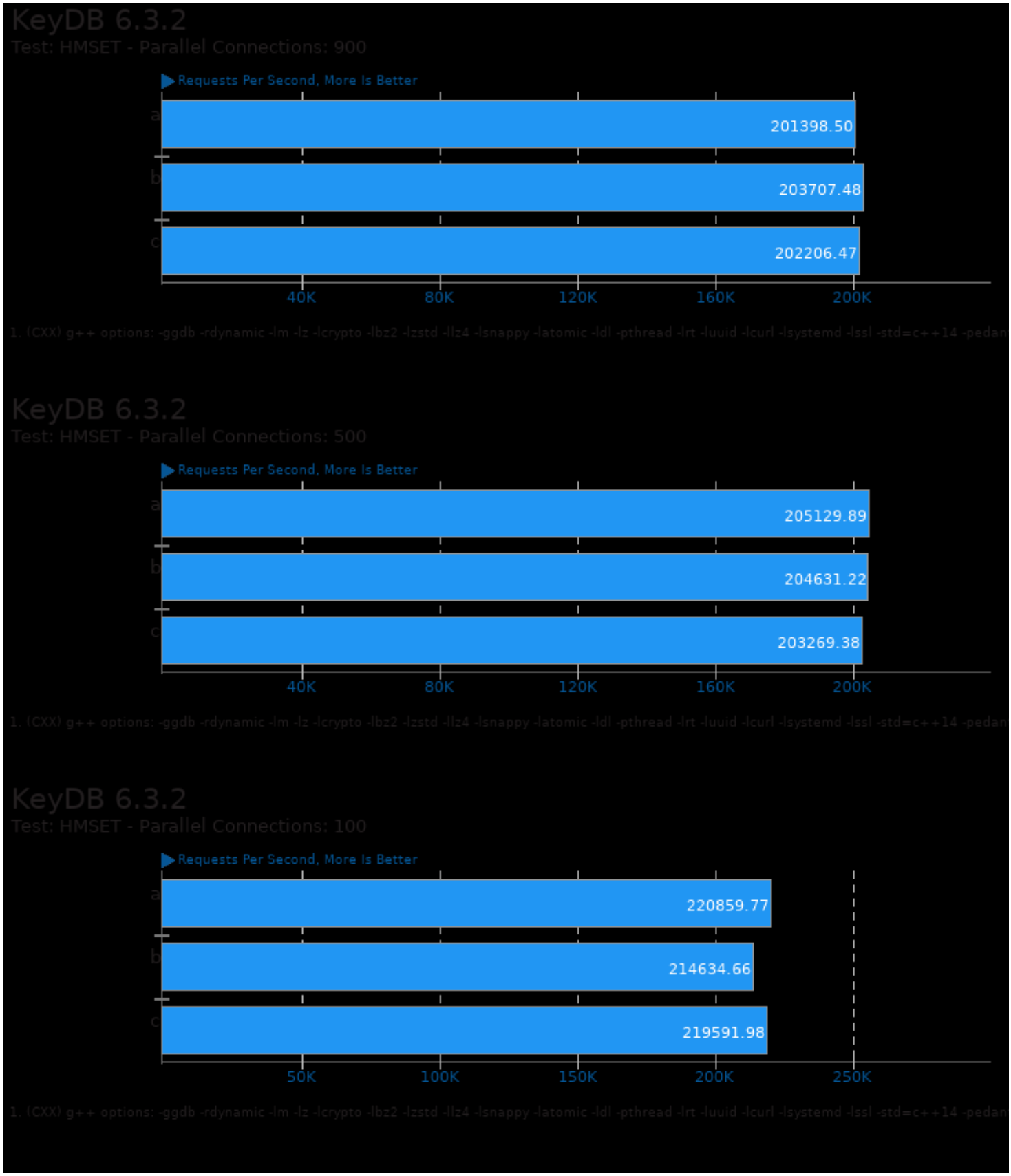
# ds lkey

| | | | |
|---|---|---|---|
| **Neural Magic DeepSparse - N.D.C.o.b.u.o.l - A.M.S (items/sec)** | 5.1659 | **5.1708** | **5.1593** |
| **Normalized** | 99.91% | 100% | 99.78% |
| **Neural Magic DeepSparse - N.T.C.B.b.u.c - A.M.S (ms/batch)** | **776.6661** | 773.8573 | **773.8149** |
| **Normalized** | 99.63% | 99.99% | 100% |
| **Neural Magic DeepSparse - N.T.C.B.b.u.c - A.M.S (items/sec)** | **5.1498** | 5.1574 | **5.169** |
| **Normalized** | 99.63% | 99.78% | 100% |
| **Neural Magic DeepSparse - C.S.9.P.Y.P - A.M.S (ms/batch)** | **563.8116** | **560.2575** | 561.5936 |
| **Normalized** | 99.37% | 100% | 99.76% |
| **Neural Magic DeepSparse - C.S.9.P.Y.P - A.M.S (items/sec)** | **7.0942** | **7.1392** | 7.1222 |
| **Normalized** | 99.37% | 100% | 99.76% |
| **Neural Magic DeepSparse - N.T.C.B.b.u.S - A.M.S (ms/batch)** | **180.2356** | **179.6011** | 180.2307 |
| **Normalized** | 99.65% | 100% | 99.65% |
| **Neural Magic DeepSparse - N.T.C.B.b.u.S - A.M.S (items/sec)** | **22.1905** | **22.2688** | 22.1912 |
| **Normalized** | 99.65% | 100% | 99.65% |
| **Neural Magic DeepSparse - N.Q.A.B.b.u.S.1.P - S.S.S (ms/batch)** | **55.2168** | 55.1247 | **54.9457** |
| **Normalized** | 99.51% | 99.68% | 100% |
| **Neural Magic DeepSparse - N.Q.A.B.b.u.S.1.P - S.S.S (items/sec)** | **18.1037** | 18.1337 | **18.1931** |
| **Normalized** | 99.51% | 99.67% | 100% |
| **Neural Magic DeepSparse - N.S.A.8.P.Q.B.B.U - S.S.S (ms/batch)** | **19.4233** | **19.2558** | 19.2663 |
| **Normalized** | 99.14% | 100% | 99.95% |
| **Neural Magic DeepSparse - N.S.A.8.P.Q.B.B.U - S.S.S (items/sec)** | **51.437** | **51.8801** | 51.8537 |
| **Normalized** | 99.15% | 100% | 99.95% |
| **Neural Magic DeepSparse - C.S.9.P.Y.P - S.S.S (ms/batch)** | **142.6454** | 142.4995 | **142.4844** |
| **Normalized** | 99.89% | 99.99% | 100% |
| **Neural Magic DeepSparse - C.S.9.P.Y.P - S.S.S (items/sec)** | **7.0091** | 7.0162 | **7.017** |
| **Normalized** | 99.89% | 99.99% | 100% |
| **Neural Magic DeepSparse - N.D.C.o.b.u.o.l - S.S.S (ms/batch)** | 198.5696 | **198.3615** | **198.8241** |
| **Normalized** | 99.9% | 100% | 99.77% |
| **Neural Magic DeepSparse - N.D.C.o.b.u.o.l - S.S.S (items/sec)** | 5.0356 | **5.0409** | **5.0291** |
| **Normalized** | 99.89% | 100% | 99.77% |
| **Neural Magic DeepSparse - N.T.C.B.b.u.c - S.S.S (ms/batch)** | **199.3423** | 199.2496 | **198.6827** |
| **Normalized** | 99.67% | 99.72% | 100% |
| **Neural Magic DeepSparse - N.T.C.B.b.u.c - S.S.S (items/sec)** | **5.0161** | 5.0184 | **5.0327** |
| **Normalized** | 99.67% | 99.72% | 100% |

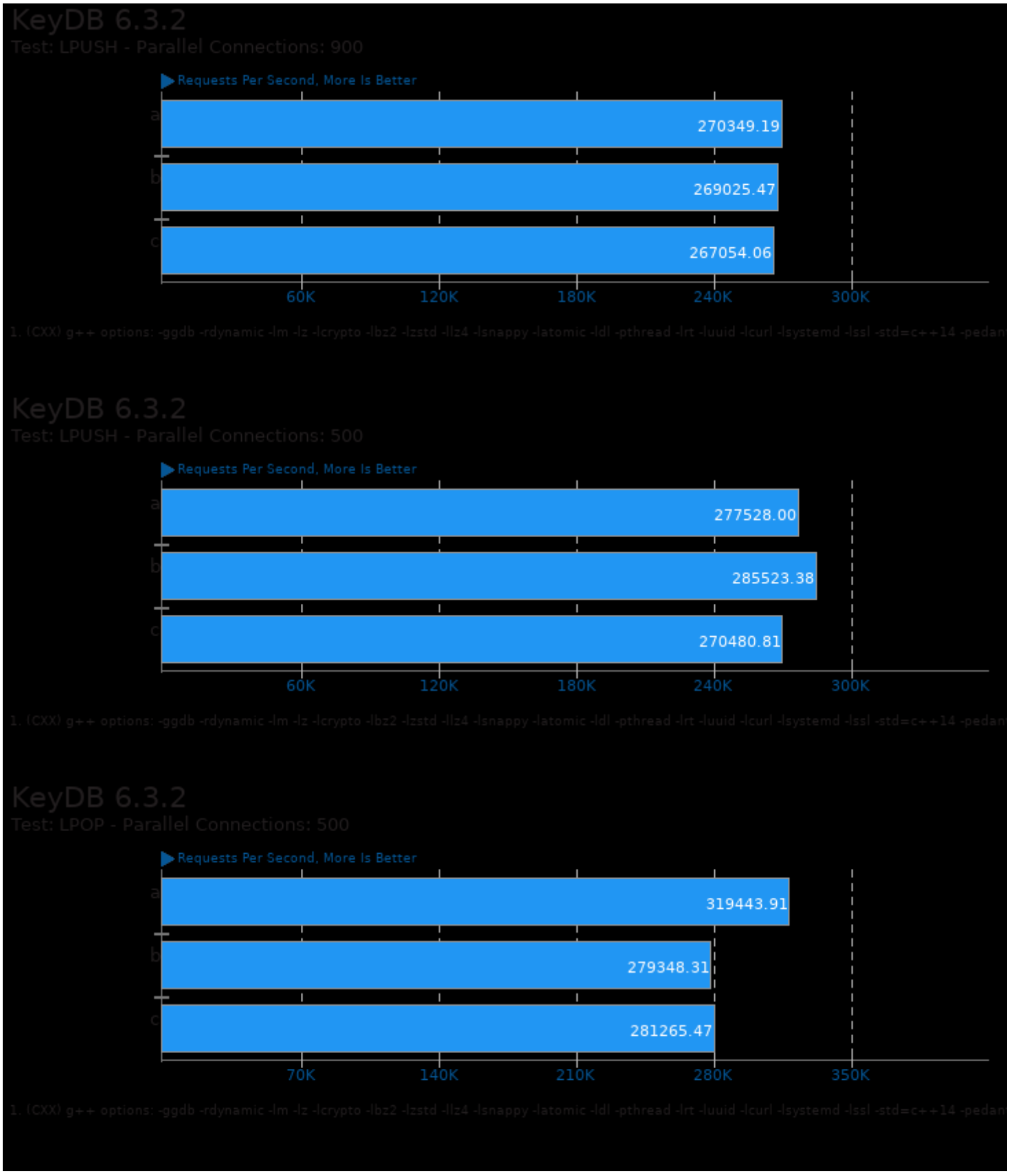
# ds lkey

| | | | |
|---|---|---|---|
| **Neural Magic DeepSparse - N.T.C.B.b.u.S - S.S.S (ms/batch)** | 49.2832 | **49.2237** | **49.4602** |
| **Normalized** | 99.88% | 100% | 99.52% |
| **Neural Magic DeepSparse - N.T.C.B.b.u.S - S.S.S (items/sec)** | 20.2841 | **20.3081** | **20.2112** |
| **Normalized** | 99.88% | 100% | 99.52% |
| **Neural Magic DeepSparse - N.T.C.D.m - A.M.S (ms/batch)** | **87.704** | 87.5767 | **87.446** |
| **Normalized** | 99.71% | 99.85% | 100% |
| **Neural Magic DeepSparse - N.T.C.D.m - A.M.S (items/sec)** | **45.5971** | 45.6629 | **45.7318** |
| **Normalized** | 99.71% | 99.85% | 100% |
| **Neural Magic DeepSparse - N.T.C.D.m - S.S.S (ms/batch)** | 24.228 | **24.2071** | **24.3051** |
| **Normalized** | 99.91% | 100% | 99.6% |
| **Neural Magic DeepSparse - N.T.C.D.m - S.S.S (items/sec)** | 41.2481 | **41.2819** | **41.1178** |
| **Normalized** | 99.92% | 100% | 99.6% |
| **Neural Magic DeepSparse - C.D.Y.C - A.M.S (ms/batch)** | **140.2665** | 140.3583 | **140.7331** |
| **Normalized** | 100% | 99.93% | 99.67% |
| **Neural Magic DeepSparse - C.D.Y.C - A.M.S (items/sec)** | **28.502** | 28.4848 | **28.4099** |
| **Normalized** | 100% | 99.94% | 99.68% |
| **Neural Magic DeepSparse - C.C.R.5.I - A.M.S (ms/batch)** | 64.3196 | **64.4599** | **64.1565** |
| **Normalized** | 99.75% | 99.53% | 100% |
| **Neural Magic DeepSparse - C.C.R.5.I - A.M.S (items/sec)** | 62.169 | **62.0323** | **62.3265** |
| **Normalized** | 99.75% | 99.53% | 100% |
| **Neural Magic DeepSparse - C.D.Y.C - S.S.S (ms/batch)** | **38.2318** | **38.1981** | 38.2315 |
| **Normalized** | 99.91% | 100% | 99.91% |
| **Neural Magic DeepSparse - C.D.Y.C - S.S.S (items/sec)** | **26.1404** | **26.1645** | 26.1412 |
| **Normalized** | 99.91% | 100% | 99.91% |
| **Neural Magic DeepSparse - C.C.R.5.I - S.S.S (ms/batch)** | 17.7369 | **17.7286** | **17.7464** |
| **Normalized** | 99.95% | 100% | 99.9% |
| **Neural Magic DeepSparse - C.C.R.5.I - S.S.S (items/sec)** | 56.3184 | **56.3495** | **56.2898** |
| **Normalized** | 99.94% | 100% | 99.89% |

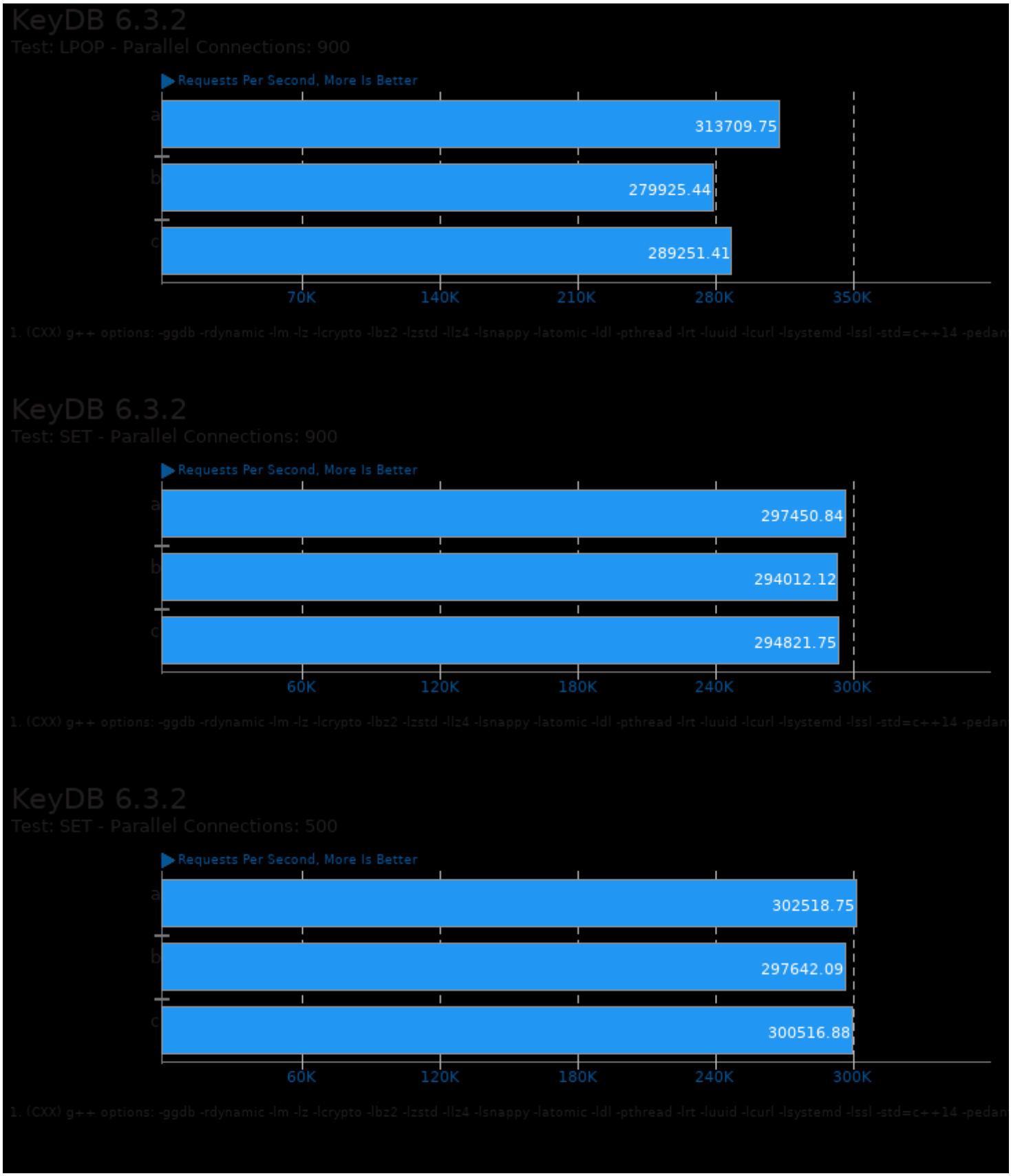
# ds lkey


KeyDB 6.3.2
Test: HMSET - Parallel Connections: 900
Requests Per Second, More Is Better
a
b
c
201398.50
203707.48
202206.47
40K
80K
120K
160K
200K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lsystemd -lssl -std=c++14 -pedan
KeyDB 6.3.2
Test: HMSET - Parallel Connections: 500
Requests Per Second, More Is Better
a
b
c
205129.89
204631.22
203269.38
40K
80K
120K
160K
200K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lsystemd -lssl -std=c++14 -pedan
KeyDB 6.3.2
Test: HMSET - Parallel Connections: 100
Requests Per Second, More Is Better
a
b
c
220859.77
214634.66
219591.98
50K
100K
150K
200K
250K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lsystemd -lssl -std=c++14 -pedan

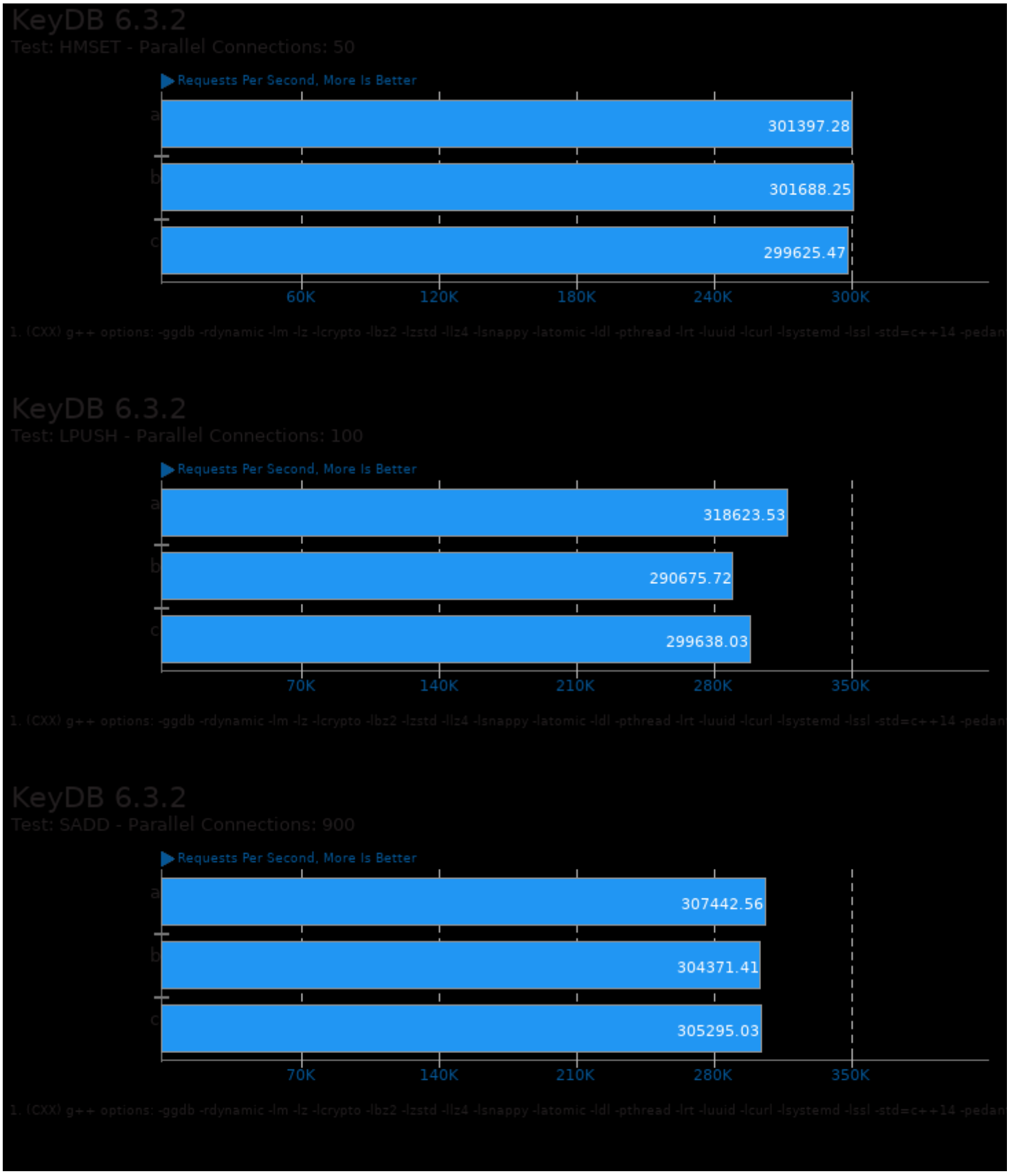
# ds lkey


KeyDB 6.3.2
Test: LPUSH - Parallel Connections: 900
Requests Per Second, More Is Better
a
270349.19
b
269025.47
c
267054.06
60K
120K
180K
240K
300K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lsystemd -lssl -std=c++14 -pedan
KeyDB 6.3.2
Test: LPUSH - Parallel Connections: 500
Requests Per Second, More Is Better
a
277528.00
b
285523.38
c
270480.81
60K
120K
180K
240K
300K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lsystemd -lssl -std=c++14 -pedan
KeyDB 6.3.2
Test: LPOP - Parallel Connections: 500
Requests Per Second, More Is Better
a
319443.91
b
279348.31
c
281265.47
70K
140K
210K
280K
350K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lsystemd -lssl -std=c++14 -pedan

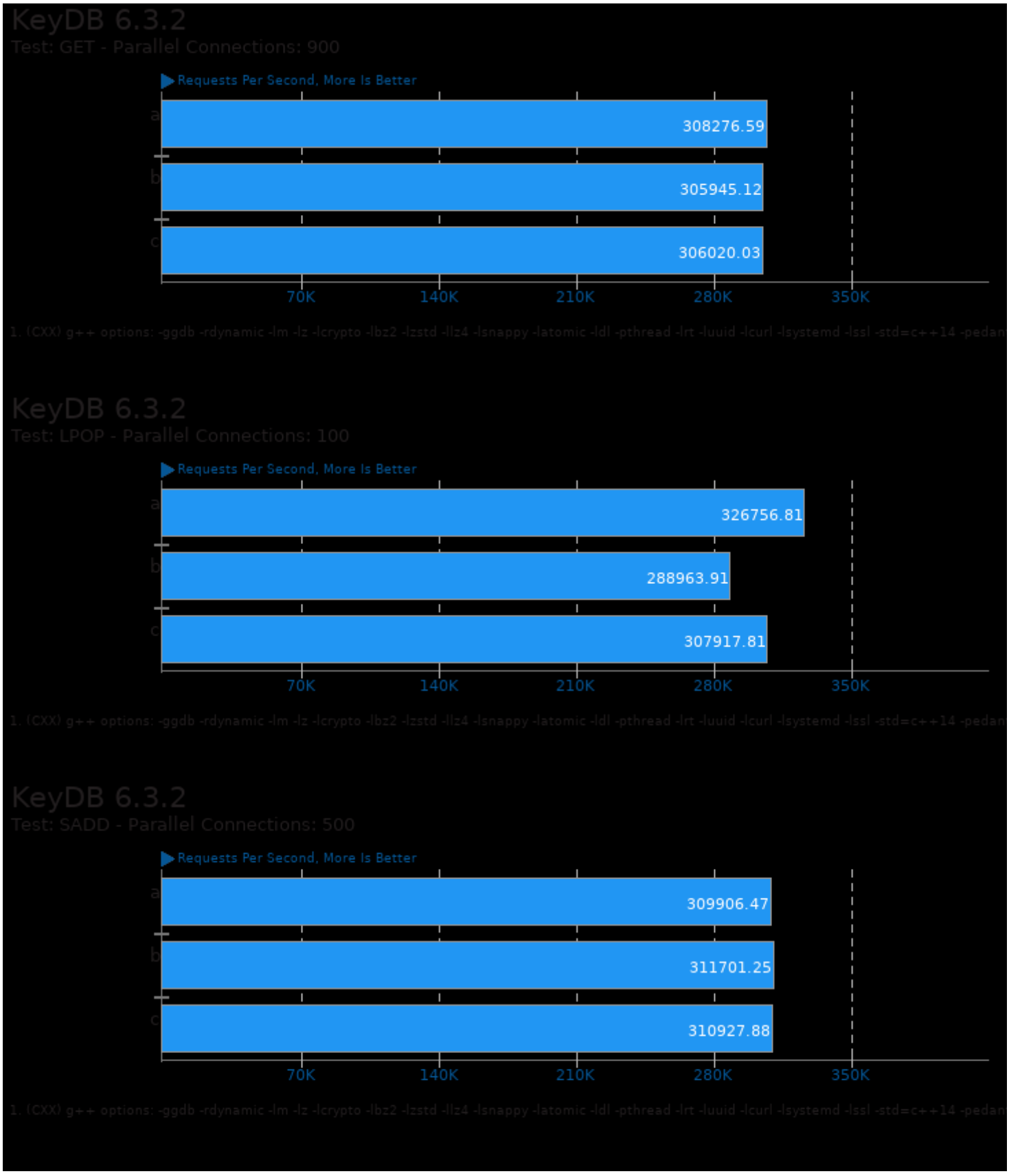
# ds lkey


KeyDB 6.3.2
Test: LPOP - Parallel Connections: 900
Requests Per Second, More Is Better
a
313709.75
b
279925.44
c
289251.41
70K
140K
210K
280K
350K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lsystemd -lssl -std=c++14 -pedan
KeyDB 6.3.2
Test: SET - Parallel Connections: 900
Requests Per Second, More Is Better
a
297450.84
b
294012.12
c
294821.75
60K
120K
180K
240K
300K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lsystemd -lssl -std=c++14 -pedan
KeyDB 6.3.2
Test: SET - Parallel Connections: 500
Requests Per Second, More Is Better
a
302518.75
b
297642.09
c
300516.88
60K
120K
180K
240K
300K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lsystemd -lssl -std=c++14 -pedan

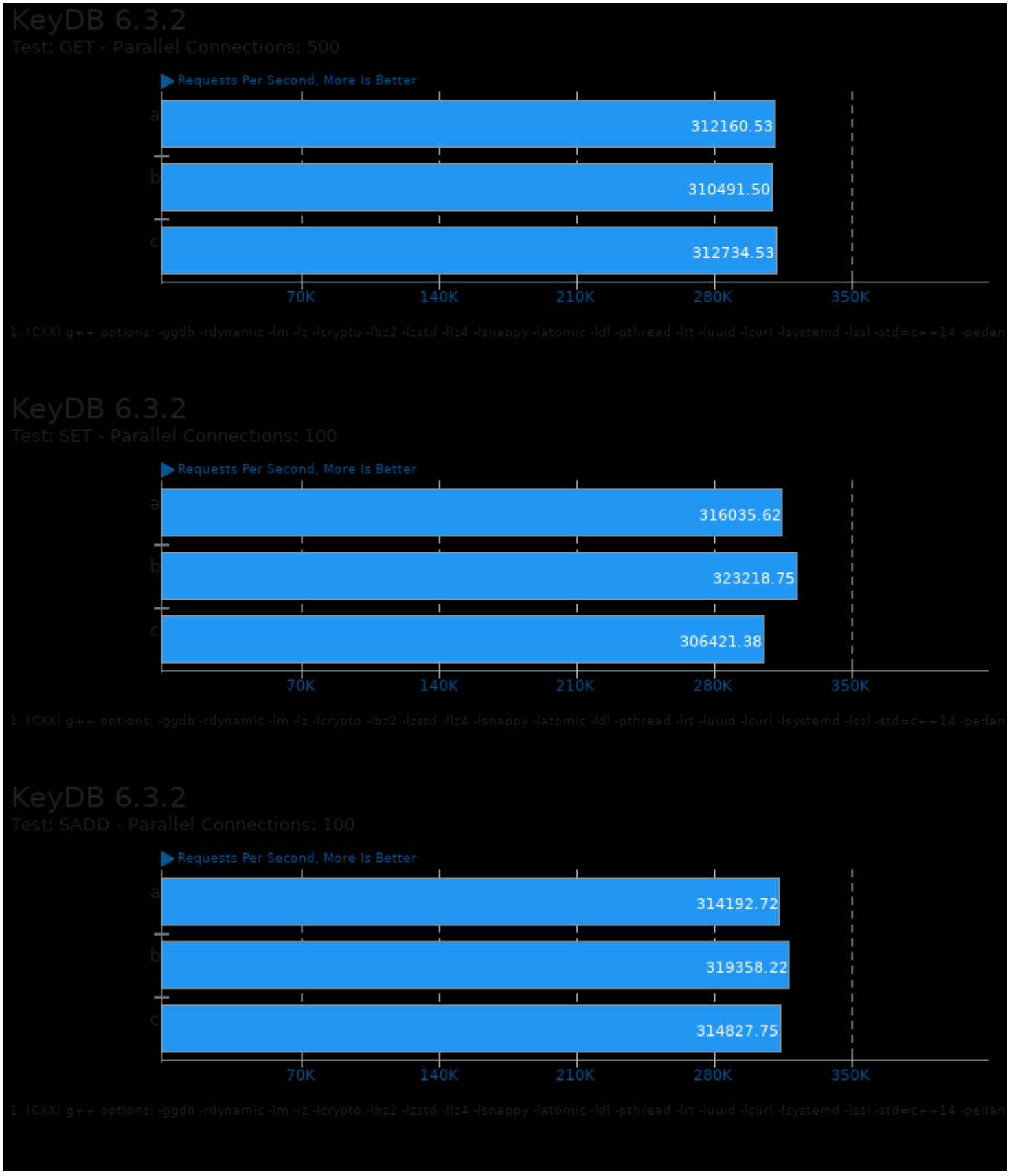
# ds lkey

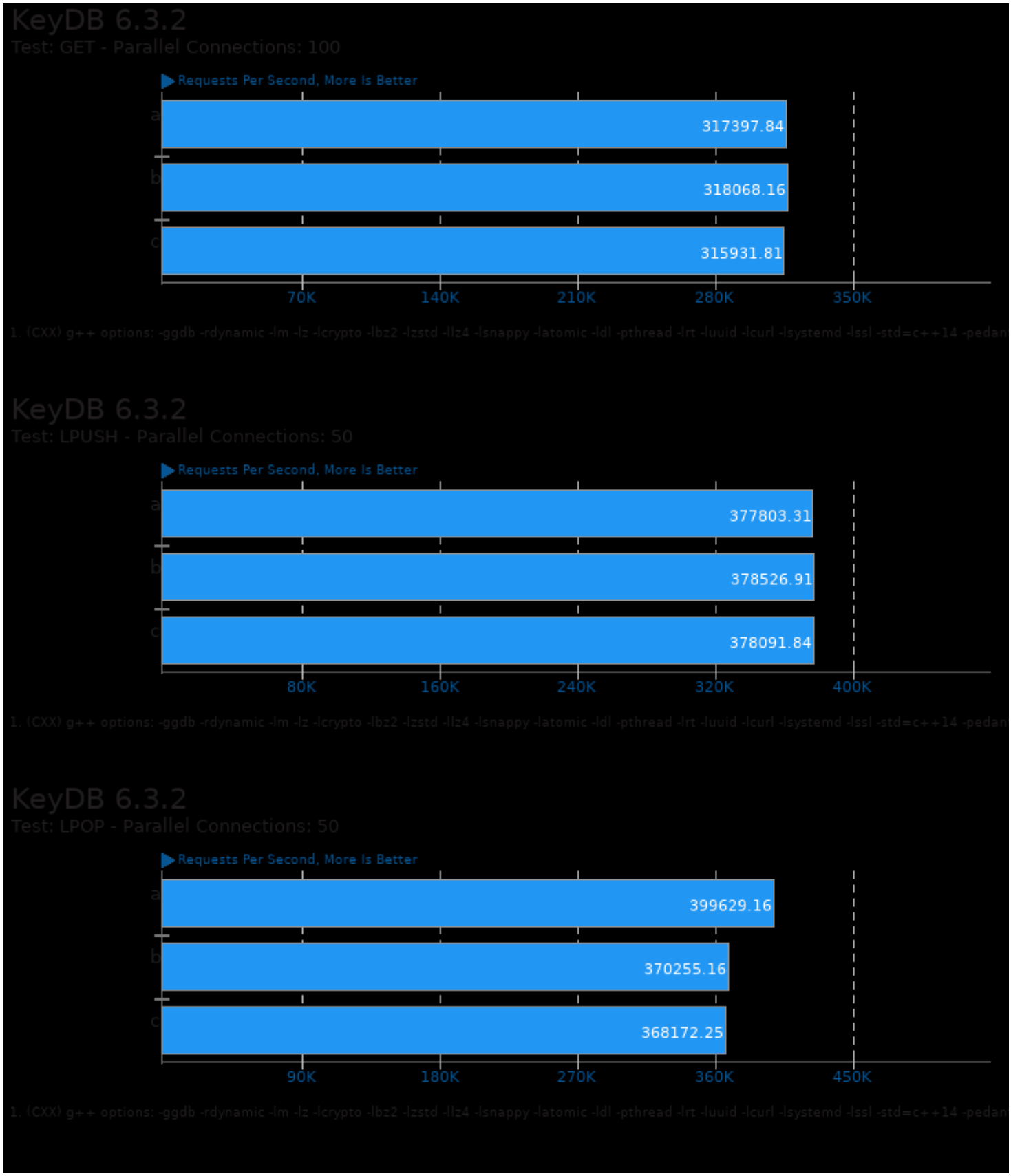

KeyDB 6.3.2
Test: HMSET - Parallel Connections: 50
Requests Per Second, More Is Better
a
301397.28
b
301688.25
c
299625.47
60K
120K
180K
240K
300K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lsystemd -lssl -std=c++14 -pedan
KeyDB 6.3.2
Test: LPUSH - Parallel Connections: 100
Requests Per Second, More Is Better
a
318623.53
b
290675.72
c
299638.03
70K
140K
210K
280K
350K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lsystemd -lssl -std=c++14 -pedan
KeyDB 6.3.2
Test: SADD - Parallel Connections: 900
Requests Per Second, More Is Better
a
307442.56
b
304371.41
c
305295.03
70K
140K
210K
280K
350K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lsystemd -lssl -std=c++14 -pedan

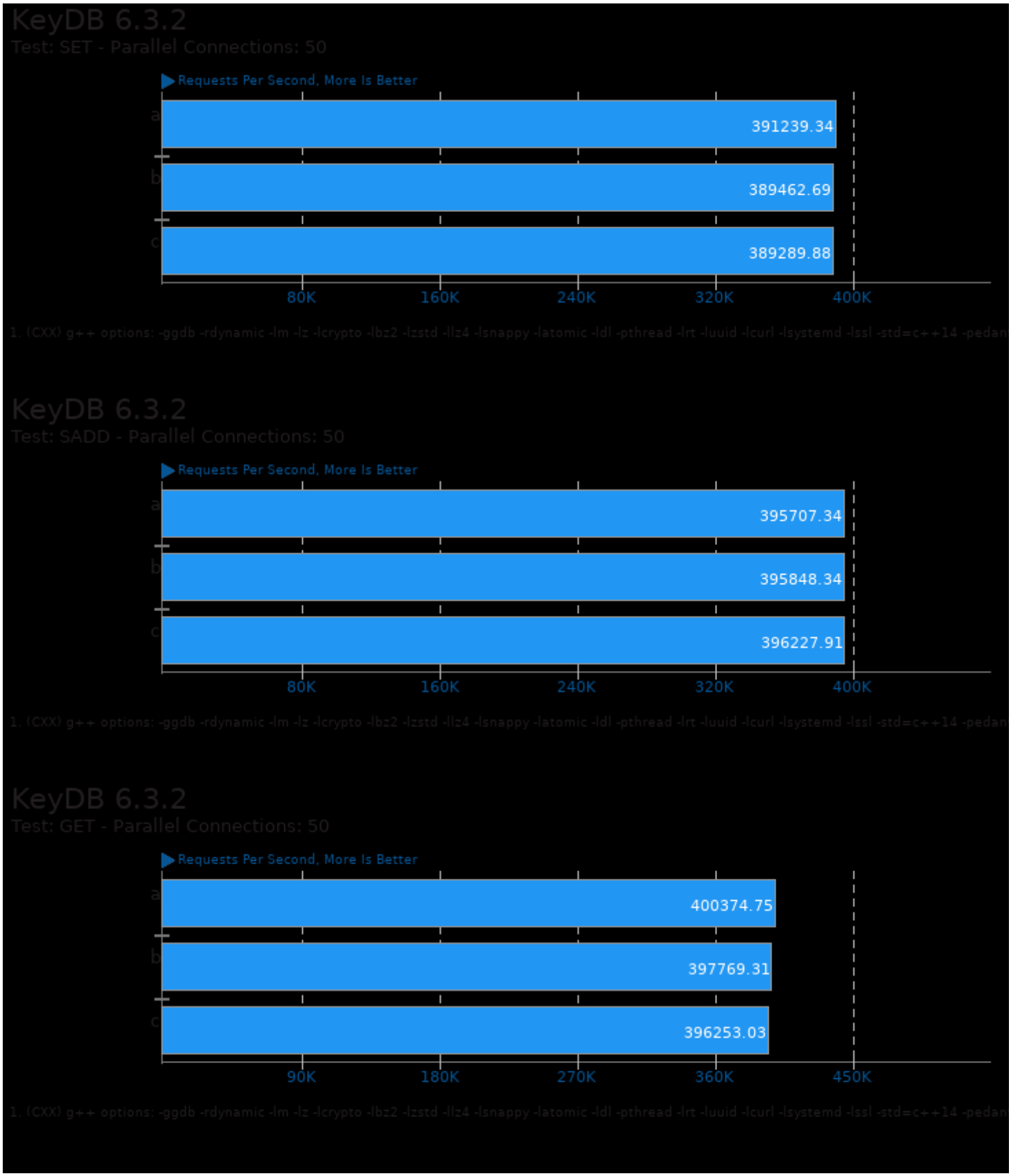
# ds lkey


KeyDB 6.3.2
Test: GET - Parallel Connections: 900
Requests Per Second, More Is Better
a
b
c
308276.59
305945.12
306020.03
70K
140K
210K
280K
350K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lsystemd -lssl -std=c++14 -pedan
KeyDB 6.3.2
Test: LPOP - Parallel Connections: 100
Requests Per Second, More Is Better
a
b
c
326756.81
288963.91
307917.81
70K
140K
210K
280K
350K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lsystemd -lssl -std=c++14 -pedan
KeyDB 6.3.2
Test: SADD - Parallel Connections: 500
Requests Per Second, More Is Better
a
b
c
309906.47
311701.25
310927.88
70K
140K
210K
280K
350K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lsystemd -lssl -std=c++14 -pedan

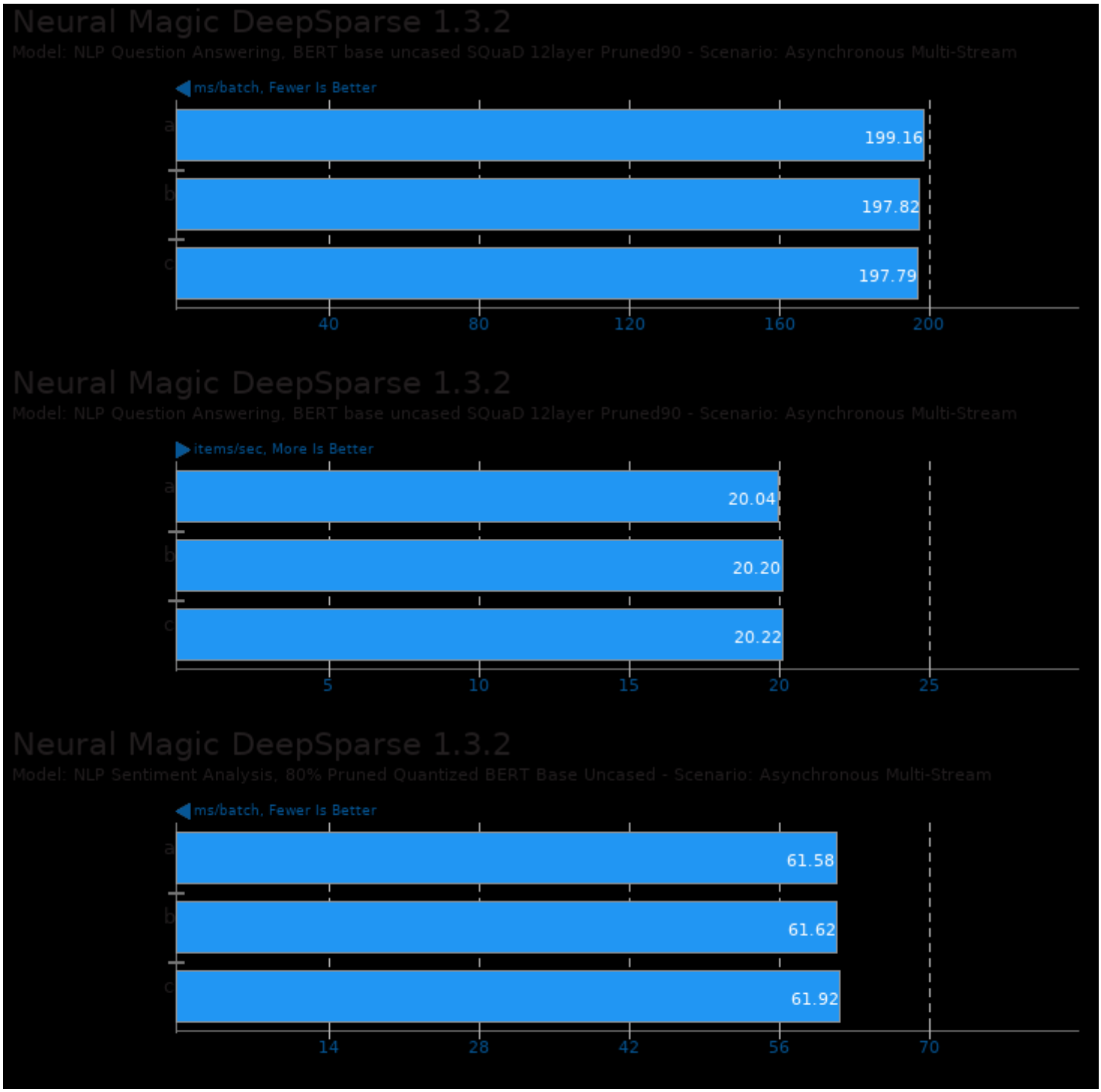
# ds lkey


KeyDB 6.3.2
Test: GET - Parallel Connections: 500
Requests Per Second, More Is Better
a
b
c
312160.53
310491.50
312734.53
70K
140K
210K
280K
350K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lsystemd -lssl -std=c++14 -pedan
KeyDB 6.3.2
Test: SET - Parallel Connections: 100
Requests Per Second, More Is Better
a
b
c
316035.62
323218.75
306421.38
70K
140K
210K
280K
350K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lsystemd -lssl -std=c++14 -pedan
KeyDB 6.3.2
Test: SADD - Parallel Connections: 100
Requests Per Second, More Is Better
a
b
c
314192.72
319358.22
314827.75
70K
140K
210K
280K
350K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lsystemd -lssl -std=c++14 -pedan

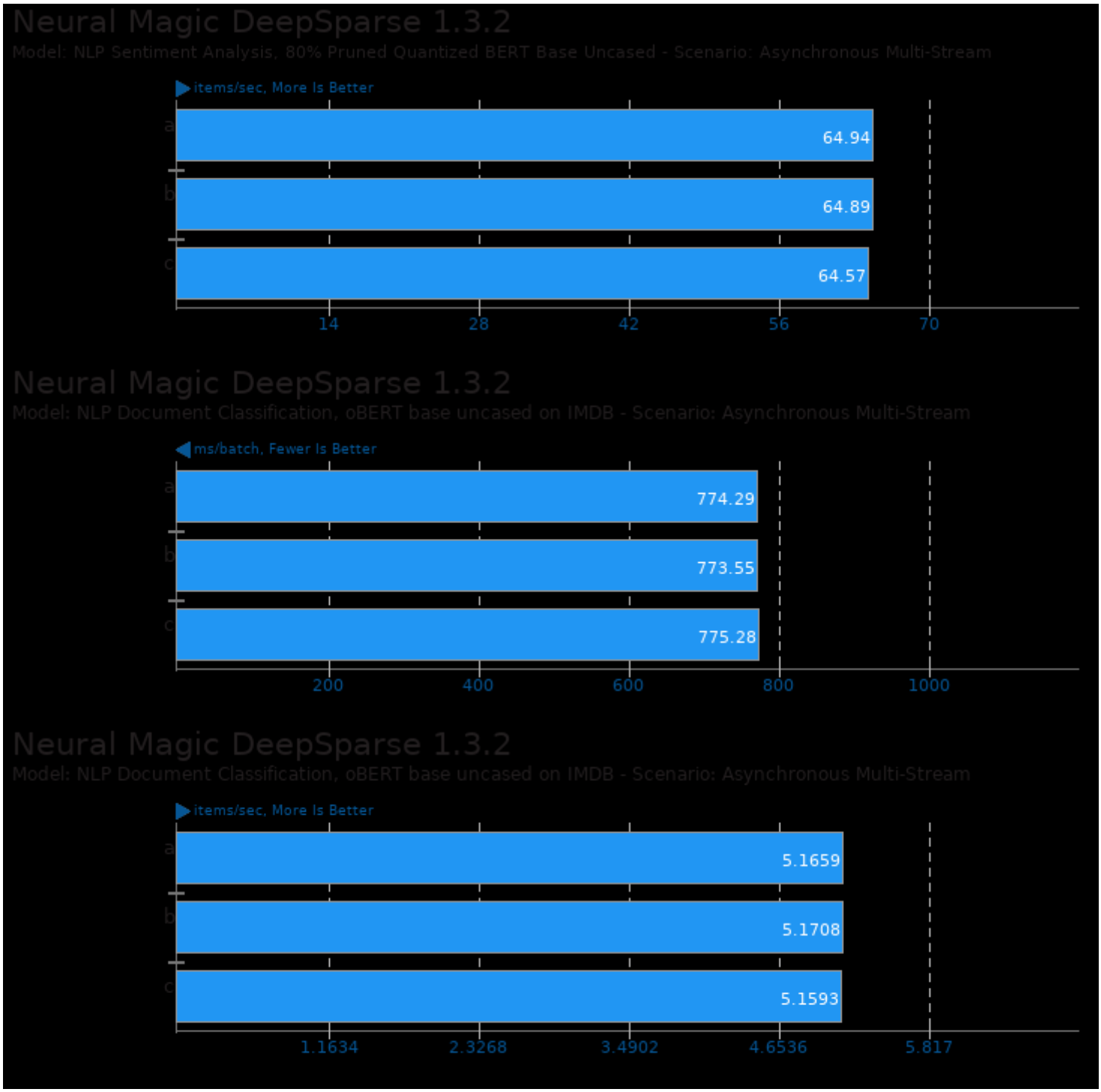

KeyDB 6.3.2
Test: GET - Parallel Connections: 100
Requests Per Second, More Is Better
a
317397.84
b
318068.16
c
315931.81
70K
140K
210K
280K
350K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lsystemd -lssl -std=c++14 -pedan
KeyDB 6.3.2
Test: LPUSH - Parallel Connections: 50
Requests Per Second, More Is Better
a
377803.31
b
378526.91
c
378091.84
80K
160K
240K
320K
400K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lsystemd -lssl -std=c++14 -pedan
KeyDB 6.3.2
Test: LPOP - Parallel Connections: 50
Requests Per Second, More Is Better
a
399629.16
b
370255.16
c
368172.25
90K
180K
270K
360K
450K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lsystemd -lssl -std=c++14 -pedan


KeyDB 6.3.2
Test: SET - Parallel Connections: 50
Requests Per Second, More Is Better
a
391239.34
b
389462.69
c
389289.88
80K
160K
240K
320K
400K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lsystemd -lssl -std=c++14 -pedan
KeyDB 6.3.2
Test: SADD - Parallel Connections: 50
Requests Per Second, More Is Better
a
395707.34
b
395848.34
c
396227.91
80K
160K
240K
320K
400K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lsystemd -lssl -std=c++14 -pedan
KeyDB 6.3.2
Test: GET - Parallel Connections: 50
Requests Per Second, More Is Better
a
400374.75
b
397769.31
c
396253.03
90K
180K
270K
360K
450K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lsystemd -lssl -std=c++14 -pedan

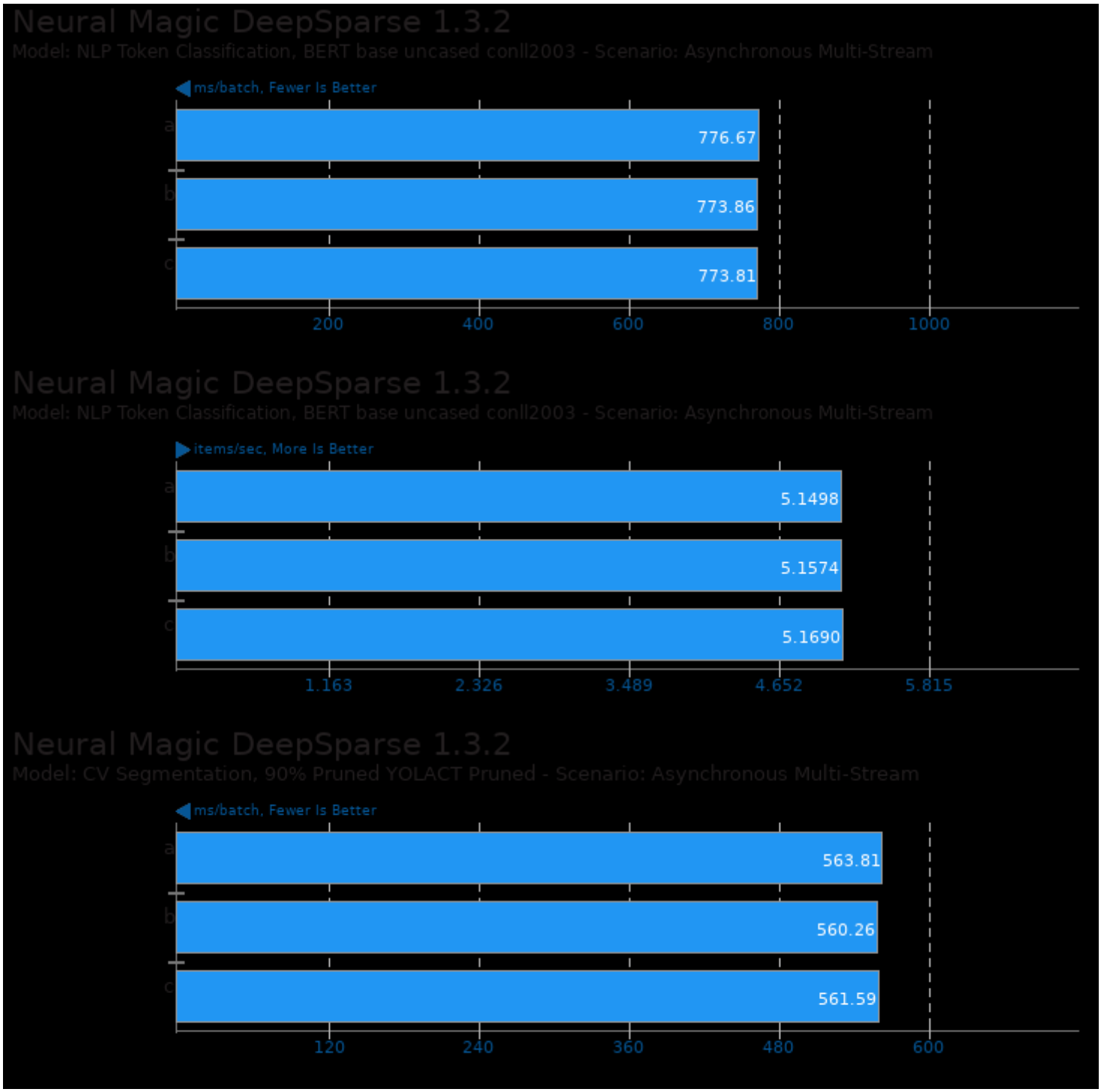
# ds lkey


Neural Magic DeepSparse 1.3.2
Model: NLP Question Answering, BERT base uncased SQuaD 12layer Pruned90 - Scenario: Asynchronous Multi-Stream
ms/batch, Fewer Is Better
a
b
c
199.16
197.82
197.79
40
80
120
160
200
Neural Magic DeepSparse 1.3.2
Model: NLP Question Answering, BERT base uncased SQuaD 12layer Pruned90 - Scenario: Asynchronous Multi-Stream
items/sec, More Is Better
a
b
c
20.04
20.20
20.22
5
10
15
20
25
Neural Magic DeepSparse 1.3.2
Model: NLP Sentiment Analysis, 80% Pruned Quantized BERT Base Uncased - Scenario: Asynchronous Multi-Stream
ms/batch, Fewer Is Better
a
b
c
61.58
61.62
61.92
14
28
42
56
70

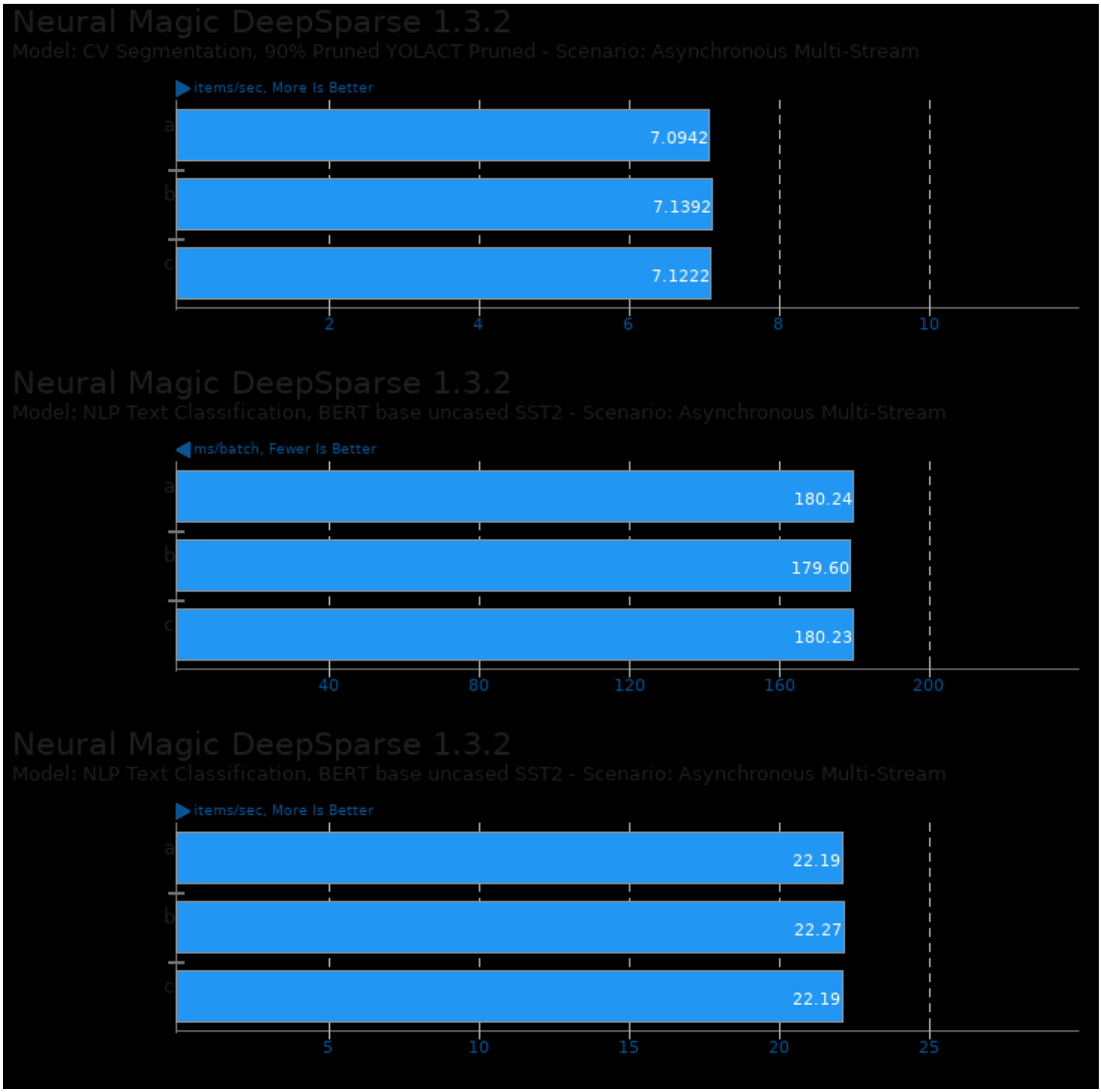
# ds lkey


Neural Magic DeepSparse 1.3.2
Model: NLP Sentiment Analysis, 80% Pruned Quantized BERT Base Uncased - Scenario: Asynchronous Multi-Stream
items/sec, More Is Better
a
b
c
64.94
64.89
64.57
14
28
42
56
70
Neural Magic DeepSparse 1.3.2
Model: NLP Document Classification, oBERT base uncased on IMDB - Scenario: Asynchronous Multi-Stream
ms/batch, Fewer Is Better
a
b
c
774.29
773.55
775.28
200
400
600
800
1000
Neural Magic DeepSparse 1.3.2
Model: NLP Document Classification, oBERT base uncased on IMDB - Scenario: Asynchronous Multi-Stream
items/sec, More Is Better
a
b
c
5.1659
5.1708
5.1593
1.1634
2.3268
3.4902
4.6536
5.817

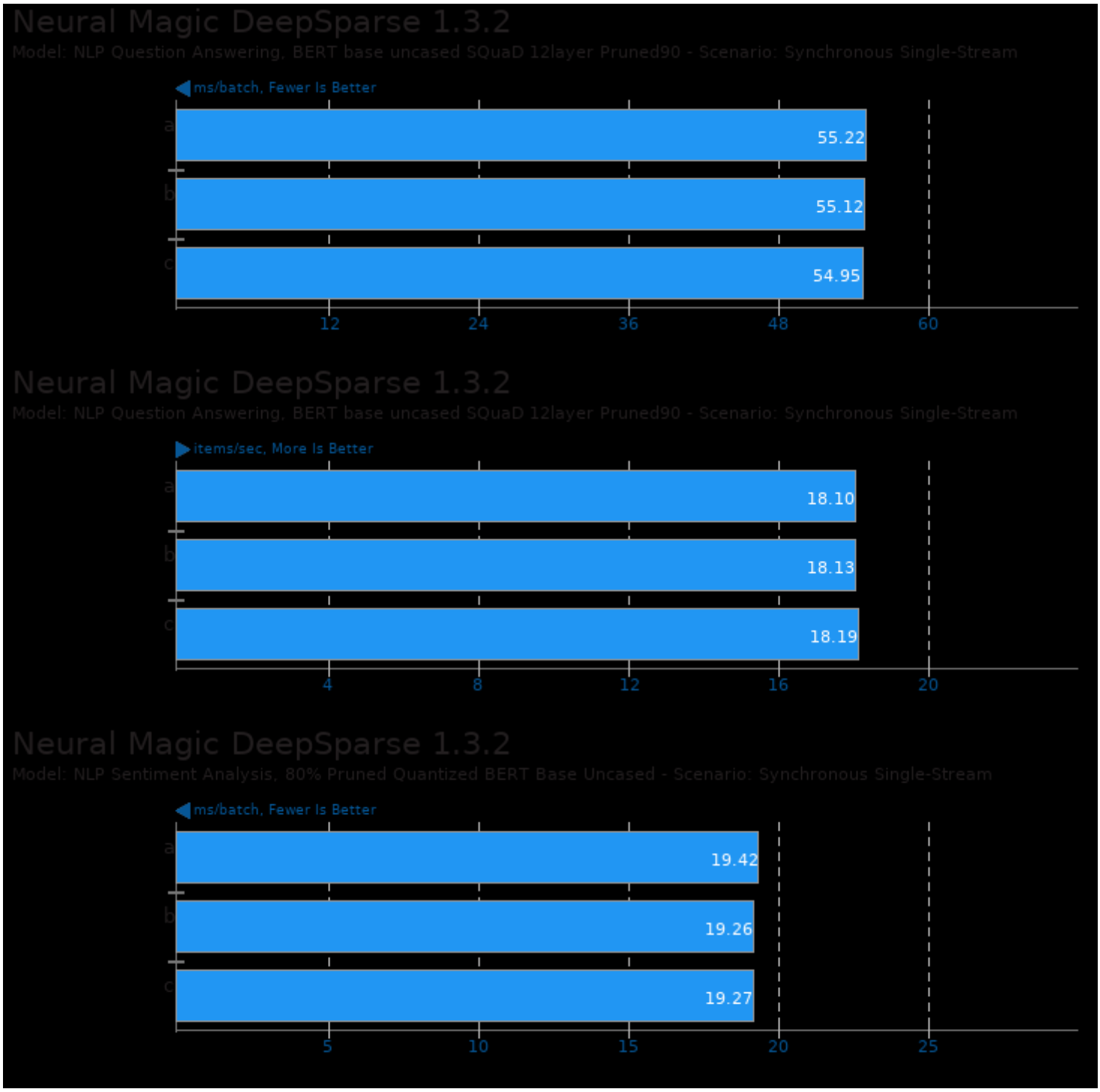
# ds lkey


Neural Magic DeepSparse 1.3.2
Model: NLP Token Classification, BERT base uncased conll2003 - Scenario: Asynchronous Multi-Stream
ms/batch, Fewer Is Better
a
b
c
776.67
773.86
773.81
200
400
600
800
1000
Neural Magic DeepSparse 1.3.2
Model: NLP Token Classification, BERT base uncased conll2003 - Scenario: Asynchronous Multi-Stream
items/sec, More Is Better
a
b
c
5.1498
5.1574
5.1690
1.163
2.326
3.489
4.652
5.815
Neural Magic DeepSparse 1.3.2
Model: CV Segmentation, 90% Pruned YOLACT Pruned - Scenario: Asynchronous Multi-Stream
ms/batch, Fewer Is Better
a
b
c
563.81
560.26
561.59
120
240
360
480
600

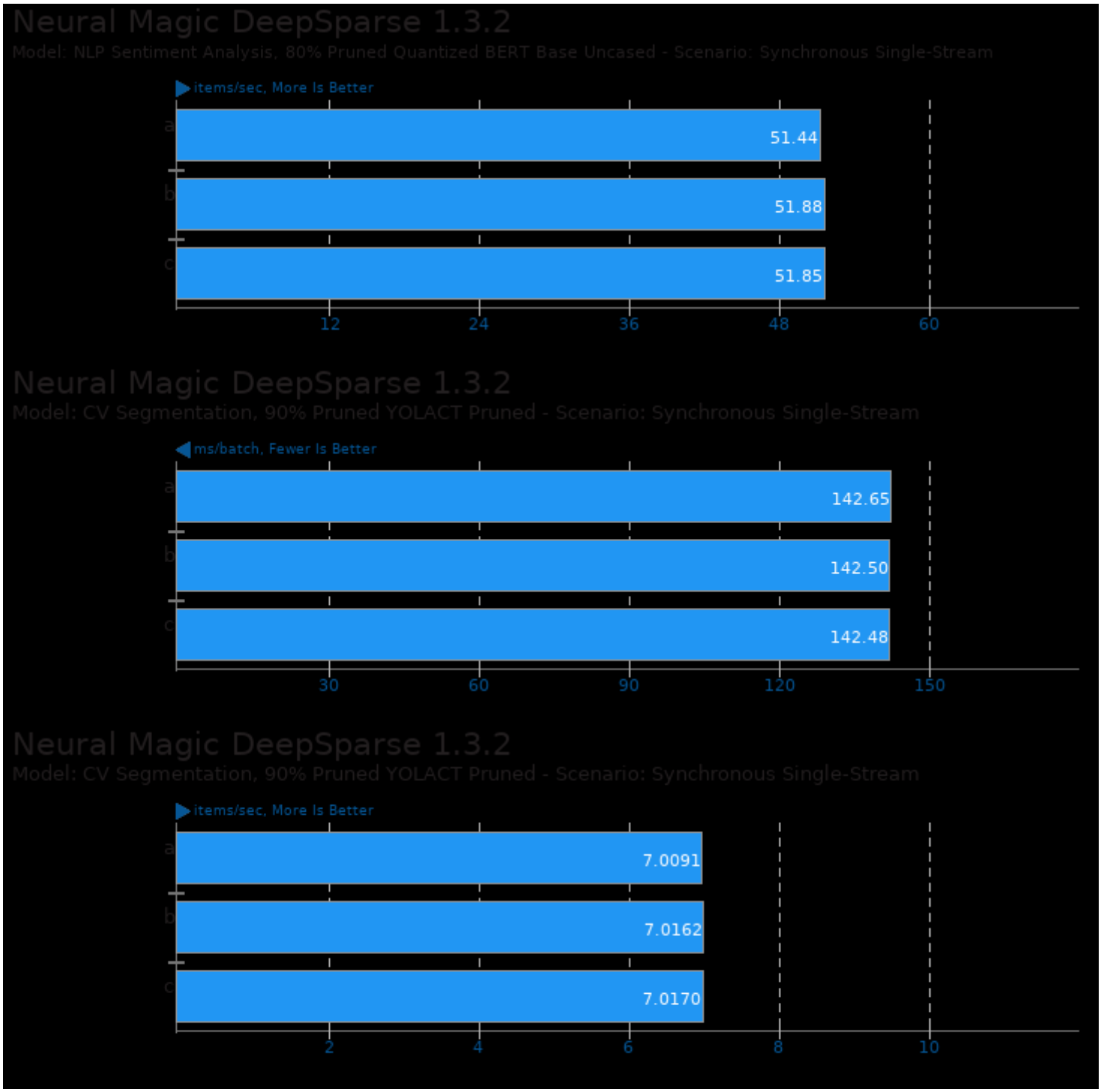
# ds lkey


Neural Magic DeepSparse 1.3.2
Model: CV Segmentation, 90% Pruned YOLACT Pruned - Scenario: Asynchronous Multi-Stream
items/sec, More Is Better
a
b
c
7.0942
7.1392
7.1222
2
4
6
8
10
Neural Magic DeepSparse 1.3.2
Model: NLP Text Classification, BERT base uncased SST2 - Scenario: Asynchronous Multi-Stream
ms/batch, Fewer Is Better
a
b
c
180.24
179.60
180.23
40
80
120
160
200
Neural Magic DeepSparse 1.3.2
Model: NLP Text Classification, BERT base uncased SST2 - Scenario: Asynchronous Multi-Stream
items/sec, More Is Better
a
b
c
22.19
22.27
22.19
5
10
15
20
25

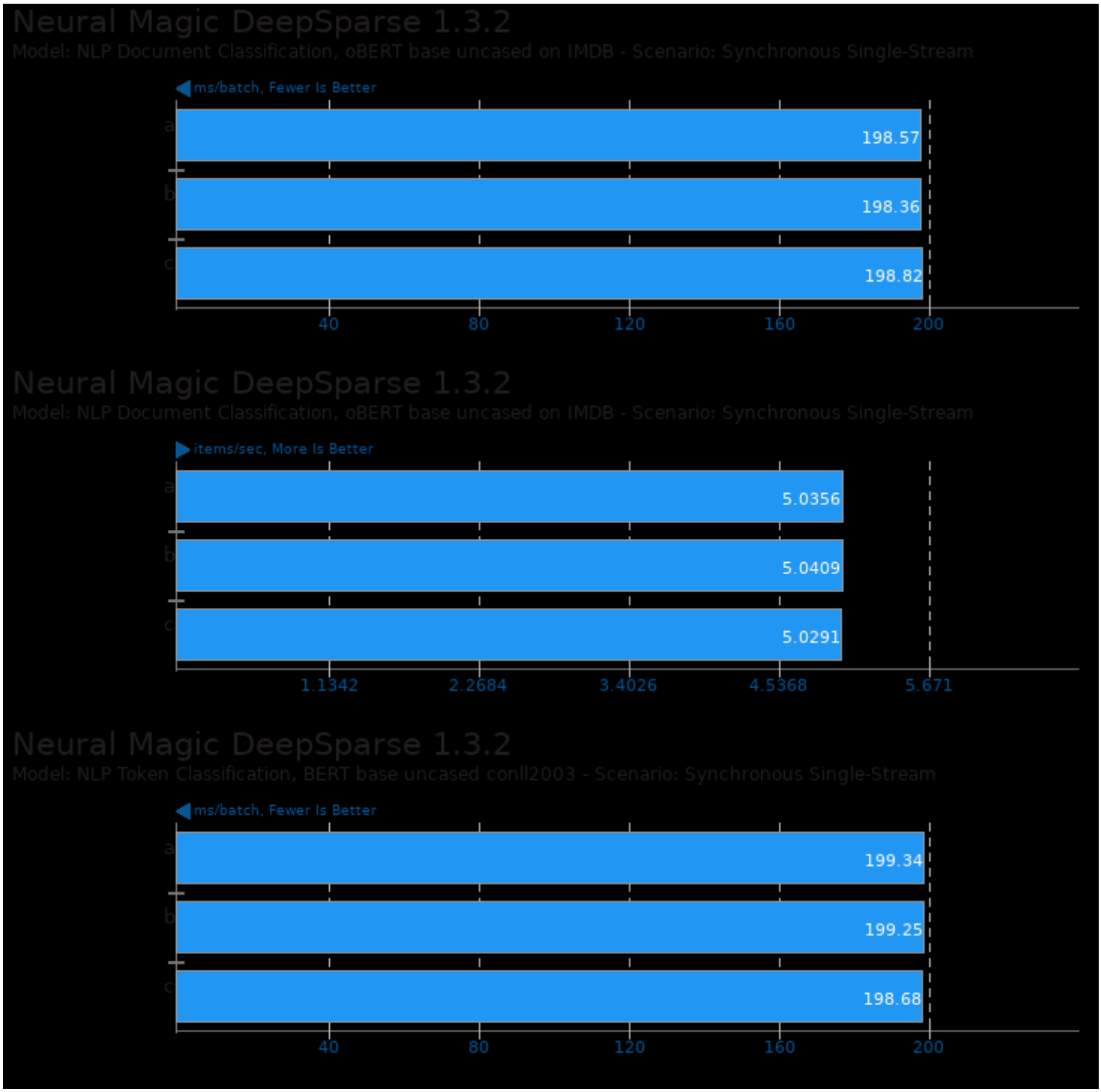
# ds lkey


Neural Magic DeepSparse 1.3.2
Model: NLP Question Answering, BERT base uncased SQuaD 12layer Pruned90 - Scenario: Synchronous Single-Stream
ms/batch, Fewer Is Better
a
b
c
55.22
55.12
54.95
12
24
36
48
60
Neural Magic DeepSparse 1.3.2
Model: NLP Question Answering, BERT base uncased SQuaD 12layer Pruned90 - Scenario: Synchronous Single-Stream
items/sec, More Is Better
a
b
c
18.10
18.13
18.19
4
8
12
16
20
Neural Magic DeepSparse 1.3.2
Model: NLP Sentiment Analysis, 80% Pruned Quantized BERT Base Uncased - Scenario: Synchronous Single-Stream
ms/batch, Fewer Is Better
a
b
c
19.42
19.26
19.27
5
10
15
20
25

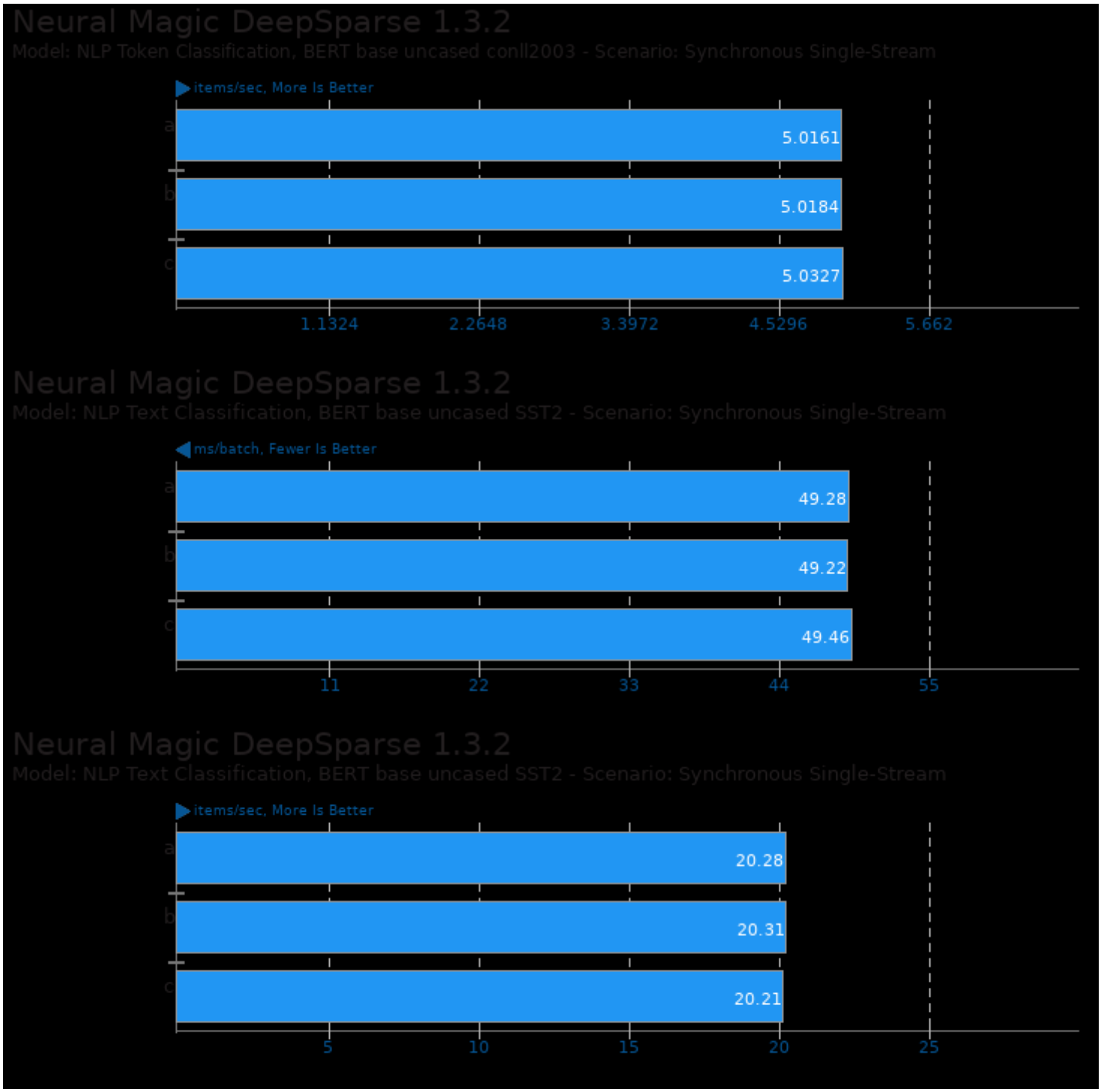
# ds lkey


Neural Magic DeepSparse 1.3.2
Model: NLP Sentiment Analysis, 80% Pruned Quantized BERT Base Uncased - Scenario: Synchronous Single-Stream
items/sec, More Is Better
a
b
c
51.44
51.88
51.85
12
24
36
48
60
Neural Magic DeepSparse 1.3.2
Model: CV Segmentation, 90% Pruned YOLACT Pruned - Scenario: Synchronous Single-Stream
ms/batch, Fewer Is Better
a
b
c
142.65
142.50
142.48
30
60
90
120
150
Neural Magic DeepSparse 1.3.2
Model: CV Segmentation, 90% Pruned YOLACT Pruned - Scenario: Synchronous Single-Stream
items/sec, More Is Better
a
b
c
7.0091
7.0162
7.0170
2
4
6
8
10

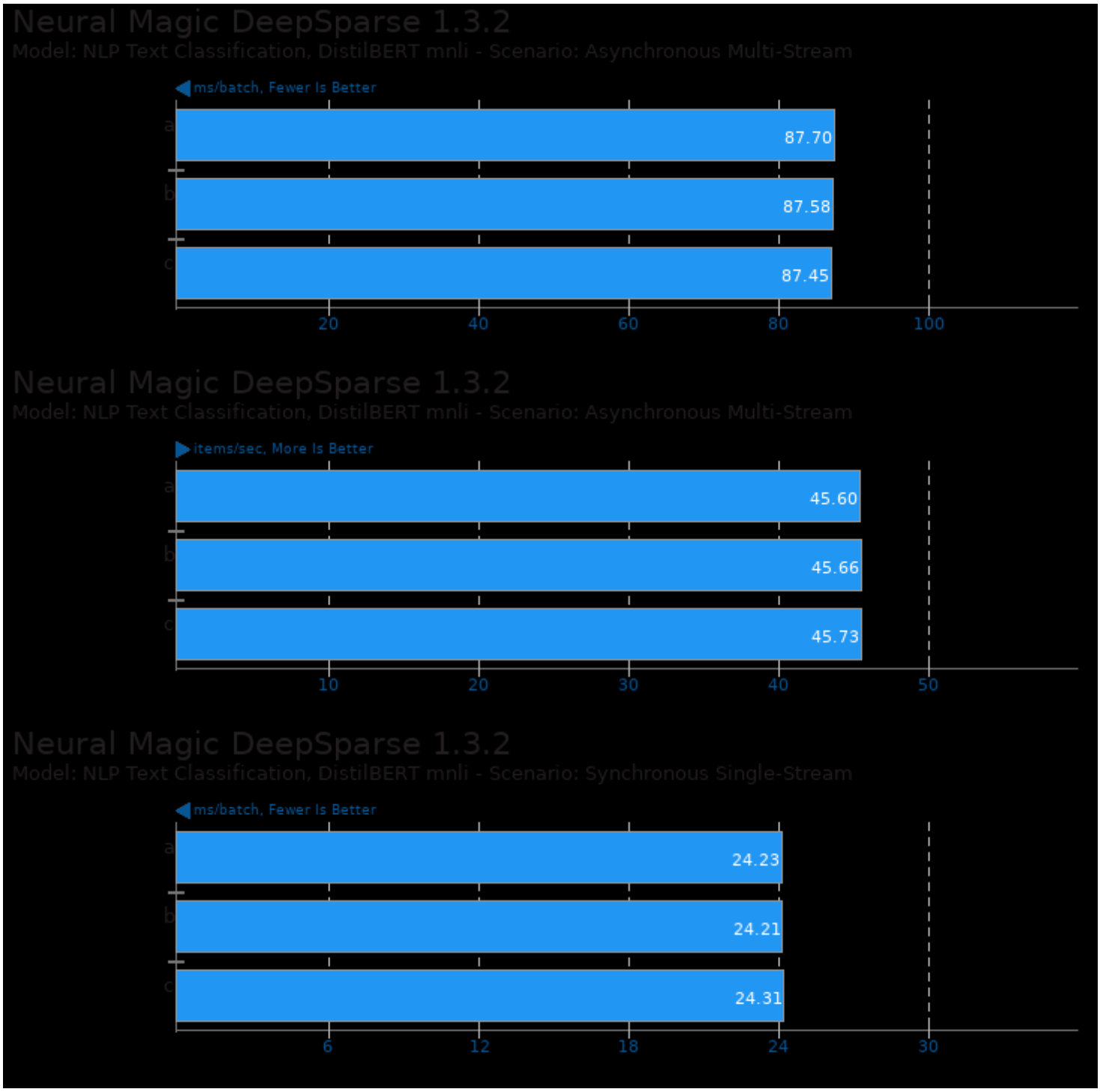
# ds lkey


Neural Magic DeepSparse 1.3.2
Model: NLP Document Classification, oBERT base uncased on IMDB - Scenario: Synchronous Single-Stream
ms/batch, Fewer Is Better
198.57
198.36
198.82
40
80
120
160
200
Neural Magic DeepSparse 1.3.2
Model: NLP Document Classification, oBERT base uncased on IMDB - Scenario: Synchronous Single-Stream
items/sec, More Is Better
5.0356
5.0409
5.0291
1.1342
2.2684
3.4026
4.5368
5.671
Neural Magic DeepSparse 1.3.2
Model: NLP Token Classification, BERT base uncased conll2003 - Scenario: Synchronous Single-Stream
ms/batch, Fewer Is Better
199.34
199.25
198.68
40
80
120
160
200

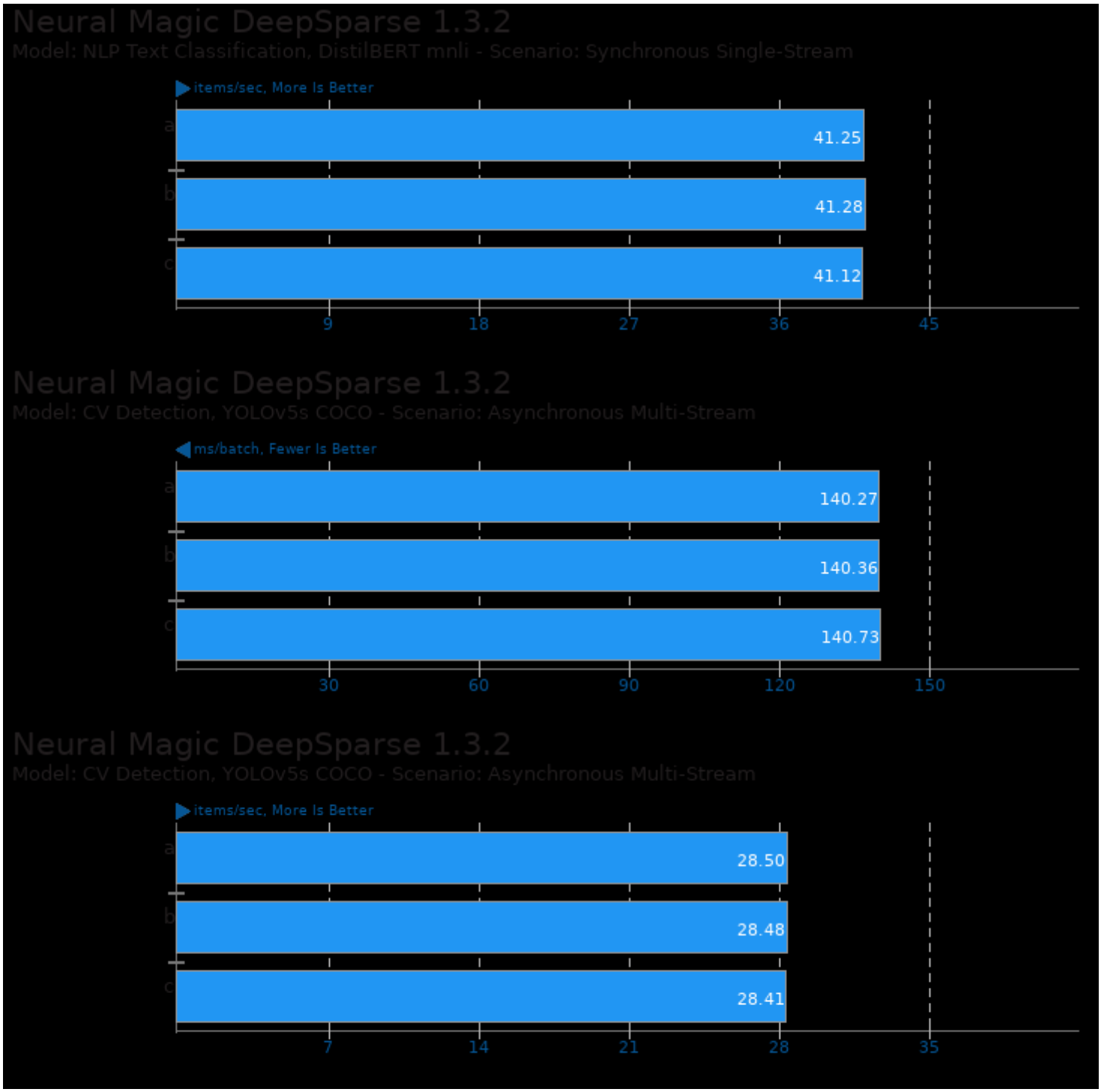
# ds lkey


Neural Magic DeepSparse 1.3.2
Model: NLP Token Classification, BERT base uncased conll2003 - Scenario: Synchronous Single-Stream
items/sec, More Is Better
a
b
c
5.0161
5.0184
5.0327
1.1324
2.2648
3.3972
4.5296
5.662
Neural Magic DeepSparse 1.3.2
Model: NLP Text Classification, BERT base uncased SST2 - Scenario: Synchronous Single-Stream
ms/batch, Fewer Is Better
a
b
c
49.28
49.22
49.46
11
22
33
44
55
Neural Magic DeepSparse 1.3.2
Model: NLP Text Classification, BERT base uncased SST2 - Scenario: Synchronous Single-Stream
items/sec, More Is Better
a
b
c
20.28
20.31
20.21
5
10
15
20
25

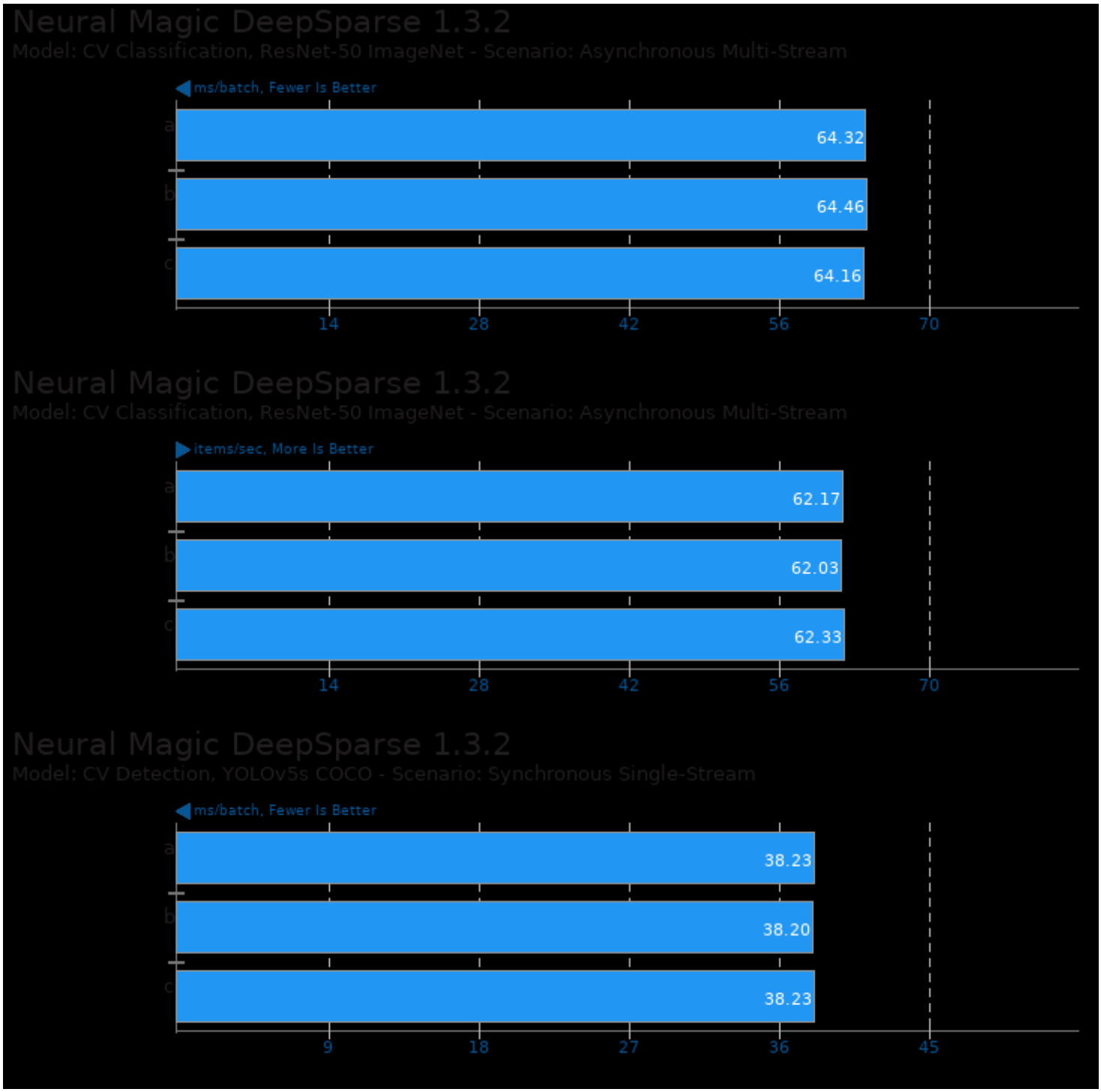
# ds lkey


Neural Magic DeepSparse 1.3.2
Model: NLP Text Classification, DistilBERT mnli - Scenario: Asynchronous Multi-Stream
ms/batch, Fewer Is Better
a
b
c
87.70
87.58
87.45
20
40
60
80
100
Neural Magic DeepSparse 1.3.2
Model: NLP Text Classification, DistilBERT mnli - Scenario: Asynchronous Multi-Stream
items/sec, More Is Better
a
b
c
45.60
45.66
45.73
10
20
30
40
50
Neural Magic DeepSparse 1.3.2
Model: NLP Text Classification, DistilBERT mnli - Scenario: Synchronous Single-Stream
ms/batch, Fewer Is Better
a
b
c
24.23
24.21
24.31
6
12
18
24
30

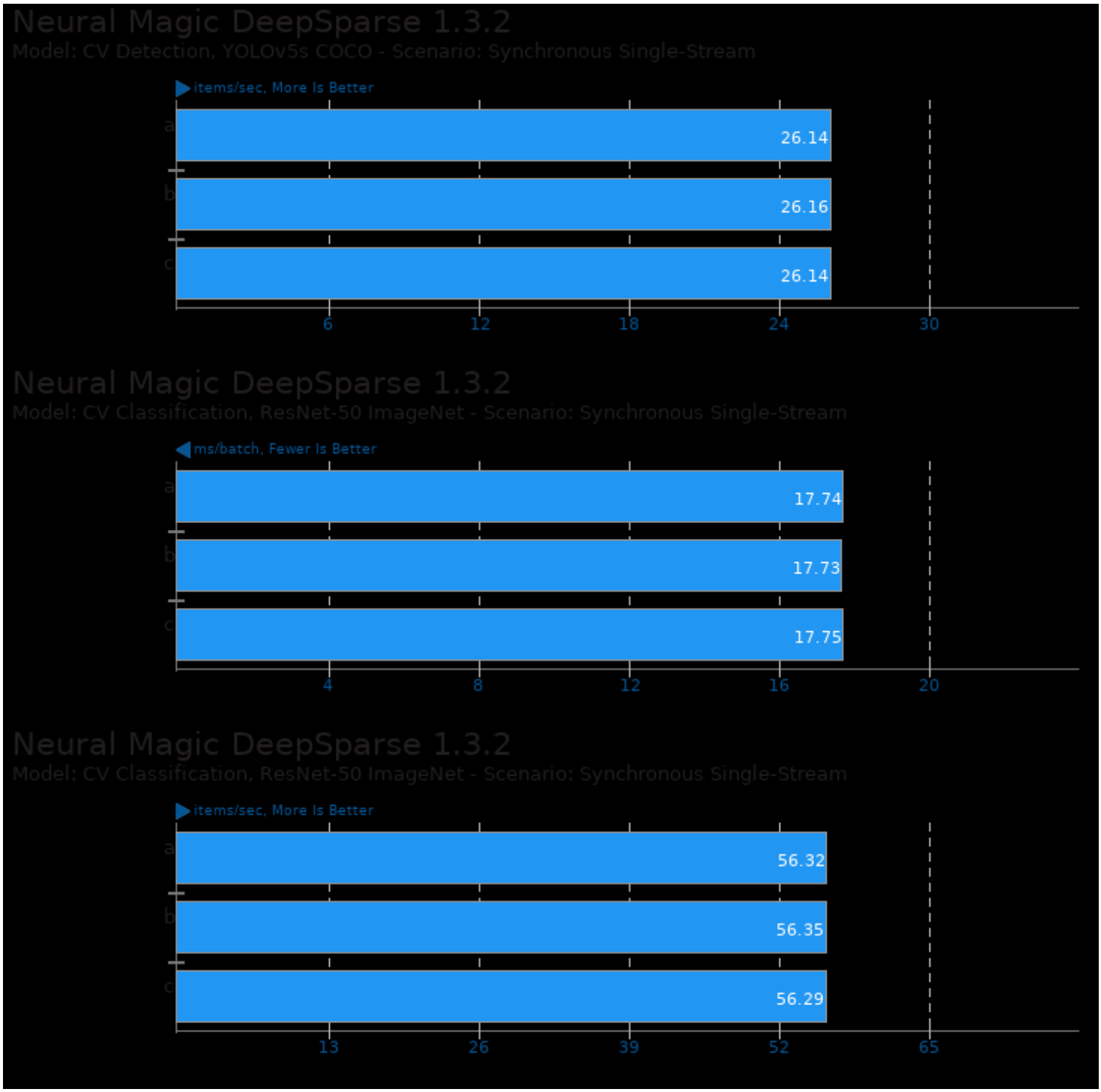
# ds lkey


Neural Magic DeepSparse 1.3.2
Model: NLP Text Classification, DistilBERT mnli - Scenario: Synchronous Single-Stream
items/sec, More Is Better
a
b
c
41.25
41.28
41.12
9
18
27
36
45
Neural Magic DeepSparse 1.3.2
Model: CV Detection, YOLOv5s COCO - Scenario: Asynchronous Multi-Stream
ms/batch, Fewer Is Better
a
b
c
140.27
140.36
140.73
30
60
90
120
150
Neural Magic DeepSparse 1.3.2
Model: CV Detection, YOLOv5s COCO - Scenario: Asynchronous Multi-Stream
items/sec, More Is Better
a
b
c
28.50
28.48
28.41
7
14
21
28
35

# ds lkey


Neural Magic DeepSparse 1.3.2
Model: CV Classification, ResNet-50 ImageNet - Scenario: Asynchronous Multi-Stream
ms/batch, Fewer Is Better
a
b
c
64.32
64.46
64.16
14
28
42
56
70
Neural Magic DeepSparse 1.3.2
Model: CV Classification, ResNet-50 ImageNet - Scenario: Asynchronous Multi-Stream
items/sec, More Is Better
a
b
c
62.17
62.03
62.33
14
28
42
56
70
Neural Magic DeepSparse 1.3.2
Model: CV Detection, YOLOv5s COCO - Scenario: Synchronous Single-Stream
ms/batch, Fewer Is Better
a
b
c
38.23
38.20
38.23
9
18
27
36
45

# ds lkey




*This file was automatically generated via the Phoronix Test Suite benchmarking software on Friday, 29 March 2024 04:42.*