


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

# ds key

Intel Core i7-1185G7 testing with a Dell 0DXP1F (3.7.0 BIOS) and Intel Xe TGL GT2 15GB on Ubuntu 22.04 via the Phoronix Test Suite.

**Automated Executive Summary**

*a had the most wins, coming in first place for 60% of the tests.*

*Based on the geometric mean of all complete results, the fastest (a) was 1.036x the speed of the slowest (c). b was 0.966x 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: 50) at 1.461x*
*KeyDB (Test: LPOP - Parallel Connections: 100) at 1.378x*
*KeyDB (Test: LPOP - Parallel Connections: 500) at 1.371x*
*KeyDB (Test: HMSET - Parallel Connections: 50) at 1.292x*
*KeyDB (Test: LPOP - Parallel Connections: 900) at 1.278x*
*KeyDB (Test: GET - Parallel Connections: 500) at 1.094x*
*KeyDB (Test: GET - Parallel Connections: 900) at 1.082x*
*KeyDB (Test: SET - Parallel Connections: 100) at 1.075x*
*Neural Magic DeepSparse (Model: NLP Text Classification, DistilBERT mnli - Scenario: Asynchronous Multi-Stream) at*

*1.075x*
*Neural Magic DeepSparse (Model: NLP Text Classification, DistilBERT mnli - Scenario: Asynchronous Multi-Stream) at 1.075x.*

# Test Systems:

## a

## b

## c

Processor: Intel Core i7-1185G7 @ 4.80GHz (4 Cores / 8 Threads), Motherboard: Dell 0DXP1F (3.7.0 BIOS), Chipset: Intel Tiger Lake-LP, Memory: 16GB, Disk: Micron 2300 NVMe 512GB, Graphics: Intel Xe TGL GT2 15GB (1350MHz), Audio: Realtek ALC289, Network: Intel Wi-Fi 6 AX201

OS: Ubuntu 22.04, Kernel: 5.18.8-051808-generic (x86_64), Desktop: GNOME Shell 42.2, Display Server: X Server + Wayland, OpenGL: 4.6 Mesa 22.0.1, OpenCL: OpenCL 3.0, Vulkan: 1.3.204, Compiler: GCC 11.3.0, 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-bootstrap --enable-cet --enable-checking=release --enable-clocale=gnu --enable-default-pie --enable-gnu-unique-object --enable-languages=c,ada,c++,go,brig,d,fortran,objc,obj-c++,m2 --enable-libphobos-checking=release --enable-libstdcxx-debug --enable-libstdcxx-time=yes --enable-link-serialization=2 --enable-multiarch --enable-multilib --enable-nls --enable-objc-gc=auto --enable-offload-targets=nvptx-none=/build/gcc-11-xKiWfi/gcc-11-11.3.0/debian/tmp-nvptx/usr,amdgcn-amdhsa=/build/gcc-11-xKiWfi/gcc-11-11.3.0/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-build-config=bootstrap-lto-lean --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: intel_pstate powersave (EPP: balance_performance) - CPU Microcode: 0xa4 - Thermald 2.4.9
Python Notes: Python 3.10.6
Security Notes: itlb_multihit: Not affected + l1tf: Not affected + mds: Not affected + meltdown: Not affected + mmio_stale_data: 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 Enhanced IBRS IBPB: conditional RSB filling + srbds: Not affected + tsx_async_abort: Not affected

| | a | b | c |
|---|---|---|---|
| **KeyDB - GET - 50 (Reqs/sec)** | **1177884** | 1135615 | **1108279** |
| **Normalized** | 100% | 96.41% | 94.09% |
| **KeyDB - SET - 50 (Reqs/sec)** | 1016157 | **1026610** | **970271** |
| **Normalized** | 98.98% | 100% | 94.51% |
| **KeyDB - GET - 100 (Reqs/sec)** | **1149108** | 1102925 | **1092729** |
| **Normalized** | 100% | 95.98% | 95.09% |
| **KeyDB - GET - 500 (Reqs/sec)** | **1142518** | **1044517** | 1108672 |
| **Normalized** | 100% | 91.42% | 97.04% |
| **KeyDB - GET - 900 (Reqs/sec)** | **1098732** | 1022516 | **1015167** |
| **Normalized** | 100% | 93.06% | 92.39% |
| **KeyDB - LPOP - 50 (Reqs/sec)** | **1237409** | **847142** | 860363 |
| **Normalized** | 100% | 68.46% | 69.53% |

# ds key

| | | | |
|---|---|---|---|
| KeyDB - SADD - 50 (Reqs/sec) | 1069336 | 1064237 | 1049935 |
| Normalized | 100% | 99.52% | 98.19% |
| KeyDB - SET - 100 (Reqs/sec) | 974545 | 923907 | 993601 |
| Normalized | 98.08% | 92.99% | 100% |
| KeyDB - SET - 500 (Reqs/sec) | 947957 | 952000 | 949884 |
| Normalized | 99.58% | 100% | 99.78% |
| KeyDB - SET - 900 (Reqs/sec) | 887233 | 877763 | 876870 |
| Normalized | 100% | 98.93% | 98.83% |
| KeyDB - HMSET - 50 (Reqs/sec) | 555130 | 516972 | 429705 |
| Normalized | 100% | 93.13% | 77.41% |
| KeyDB - LPOP - 100 (Reqs/sec) | 1164497 | 844894 | 855593 |
| Normalized | 100% | 72.55% | 73.47% |
| KeyDB - LPOP - 500 (Reqs/sec) | 1187254 | 869338 | 865951 |
| Normalized | 100% | 73.22% | 72.94% |
| KeyDB - LPOP - 900 (Reqs/sec) | 1093757 | 859048 | 855915 |
| Normalized | 100% | 78.54% | 78.25% |
| KeyDB - LPUSH - 50 (Reqs/sec) | 873561 | 878488 | 873134 |
| Normalized | 99.44% | 100% | 99.39% |
| KeyDB - SADD - 100 (Reqs/sec) | 1056636 | 1069153 | 1078702 |
| Normalized | 97.95% | 99.11% | 100% |
| KeyDB - SADD - 500 (Reqs/sec) | 1049120 | 979432 | 1019514 |
| Normalized | 100% | 93.36% | 97.18% |
| KeyDB - SADD - 900 (Reqs/sec) | 971534 | 982241 | 987869 |
| Normalized | 98.35% | 99.43% | 100% |
| KeyDB - HMSET - 100 (Reqs/sec) | 351400 | 340382 | 333220 |
| Normalized | 100% | 96.86% | 94.83% |
| KeyDB - HMSET - 500 (Reqs/sec) | 339833 | 331139 | 333227 |
| Normalized | 100% | 97.44% | 98.06% |
| KeyDB - HMSET - 900 (Reqs/sec) | 334560 | 316316 | 324929 |
| Normalized | 100% | 94.55% | 97.12% |
| KeyDB - LPUSH - 100 (Reqs/sec) | 835324 | 846181 | 809887 |
| Normalized | 98.72% | 100% | 95.71% |
| KeyDB - LPUSH - 500 (Reqs/sec) | 789391 | 800192 | 785016 |
| Normalized | 98.65% | 100% | 98.1% |
| KeyDB - LPUSH - 900 (Reqs/sec) | 745145 | 727590 | 725637 |
| Normalized | 100% | 97.64% | 97.38% |
| Neural Magic DeepSparse - N.D.C.o.b.u.o.l - A.M.S (items/sec) | 3.6432 | 3.4334 | 3.5142 |
| Normalized | 100% | 94.24% | 96.46% |
| Neural Magic DeepSparse - N.D.C.o.b.u.o.l - A.M.S (ms/batch) | 548.9192 | 582.4832 | 569.0946 |
| Normalized | 100% | 94.24% | 96.45% |
| Neural Magic DeepSparse - N.D.C.o.b.u.o.l - S.S.S (items/sec) | 3.2019 | 3.1968 | 3.1813 |
| Normalized | 100% | 99.84% | 99.36% |
| Neural Magic DeepSparse - N.D.C.o.b.u.o.l - S.S.S (ms/batch) | 312.3036 | 312.806 | 314.331 |
| Normalized | 100% | 99.84% | 99.36% |
| Neural Magic DeepSparse - N.S.A.8.P.Q.B.B.U - A.M.S (items/sec) | 64.2481 | 63.9514 | 63.7129 |
| Normalized | 100% | 99.54% | 99.17% |

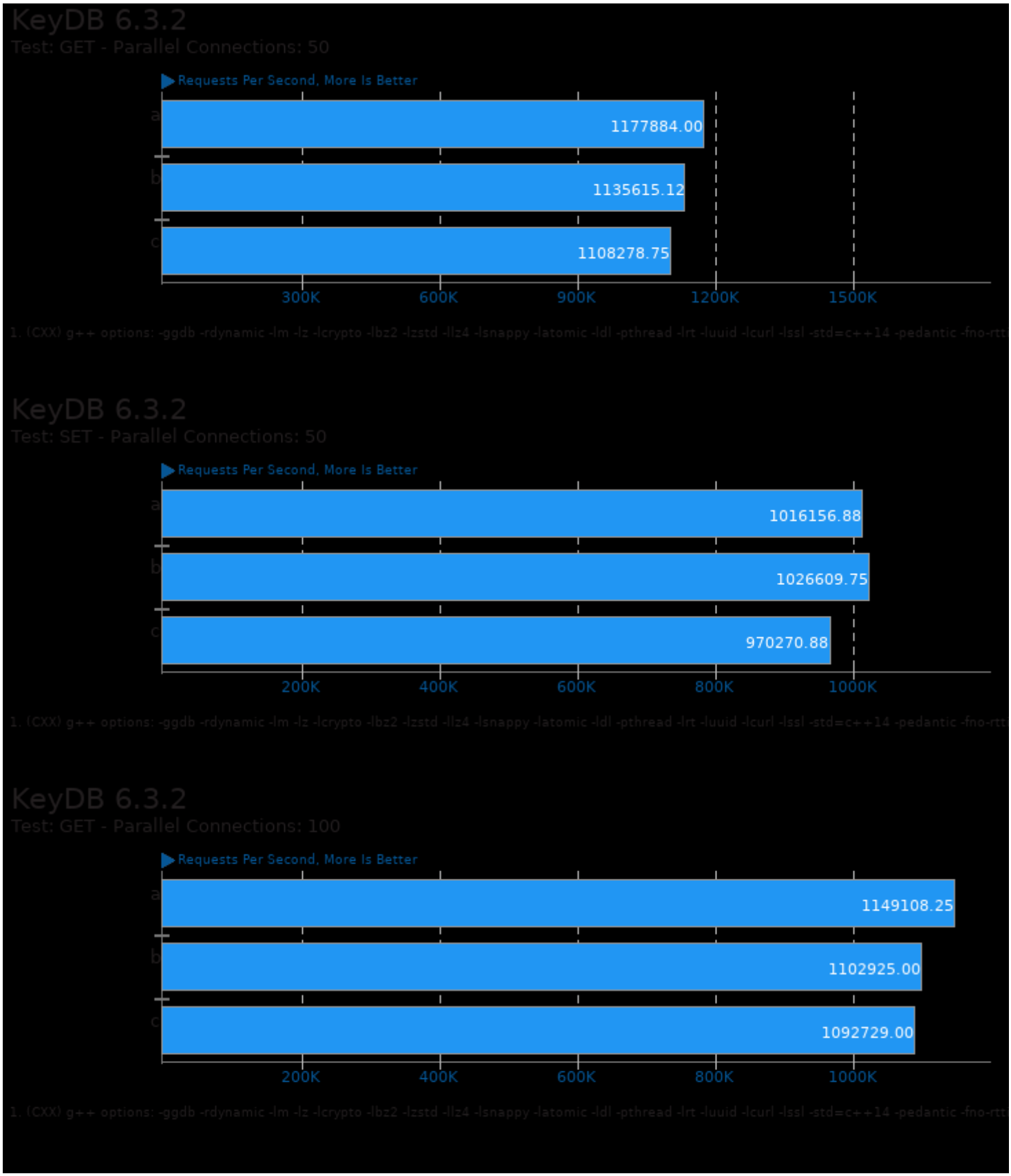
# ds key

| | | | |
|---|---|---|---|
| Neural Magic DeepSparse - N.S.A.8.P.Q.B.B.U - A.M.S (ms/batch) | 31.09 | 31.2321 | 31.3553 |
| Normalized | 100% | 99.55% | 99.15% |
| Neural Magic DeepSparse - N.S.A.8.P.Q.B.B.U - S.S.S (items/sec) | 57.3951 | 57.6809 | 57.5623 |
| Normalized | 99.5% | 100% | 99.79% |
| Neural Magic DeepSparse - N.S.A.8.P.Q.B.B.U - S.S.S (ms/batch) | 17.4128 | 17.3259 | 17.3623 |
| Normalized | 99.5% | 100% | 99.79% |
| Neural Magic DeepSparse - N.Q.A.B.b.u.S.1.P - A.M.S (items/sec) | 16.348 | 16.3411 | 16.2252 |
| Normalized | 100% | 99.96% | 99.25% |
| Neural Magic DeepSparse - N.Q.A.B.b.u.S.1.P - A.M.S (ms/batch) | 122.2412 | 122.3064 | 123.2349 |
| Normalized | 100% | 99.95% | 99.19% |
| Neural Magic DeepSparse - N.Q.A.B.b.u.S.1.P - S.S.S (items/sec) | 16.7654 | 16.6946 | 16.7245 |
| Normalized | 100% | 99.58% | 99.76% |
| Neural Magic DeepSparse - N.Q.A.B.b.u.S.1.P - S.S.S (ms/batch) | 59.6361 | 59.8888 | 59.7819 |
| Normalized | 100% | 99.58% | 99.76% |
| Neural Magic DeepSparse - C.D.Y.C - A.M.S (items/sec) | 20.9532 | 20.9856 | 20.954 |
| Normalized | 99.85% | 100% | 99.85% |
| Neural Magic DeepSparse - C.D.Y.C - A.M.S (ms/batch) | 95.4102 | 95.2684 | 95.4109 |
| Normalized | 99.85% | 100% | 99.85% |
| Neural Magic DeepSparse - C.D.Y.C - S.S.S (items/sec) | 18.2894 | 18.3035 | 18.2301 |
| Normalized | 99.92% | 100% | 99.6% |
| Neural Magic DeepSparse - C.D.Y.C - S.S.S (ms/batch) | 54.6625 | 54.6207 | 54.8399 |
| Normalized | 99.92% | 100% | 99.6% |
| Neural Magic DeepSparse - C.C.R.5.I - A.M.S (items/sec) | 53.2299 | 53.3125 | 53.4162 |
| Normalized | 99.65% | 99.81% | 100% |
| Neural Magic DeepSparse - C.C.R.5.I - A.M.S (ms/batch) | 37.5301 | 37.4795 | 37.4021 |
| Normalized | 99.66% | 99.79% | 100% |
| Neural Magic DeepSparse - C.C.R.5.I - S.S.S (items/sec) | 51.5187 | 51.5115 | 51.4994 |
| Normalized | 100% | 99.99% | 99.96% |
| Neural Magic DeepSparse - C.C.R.5.I - S.S.S (ms/batch) | 19.399 | 19.4024 | 19.4068 |
| Normalized | 100% | 99.98% | 99.96% |
| Neural Magic DeepSparse - N.T.C.D.m - A.M.S (items/sec) | 29.6472 | 27.5874 | 29.3697 |
| Normalized | 100% | 93.05% | 99.06% |
| Neural Magic DeepSparse - N.T.C.D.m - A.M.S (ms/batch) | 67.4364 | 72.4726 | 68.0739 |
| Normalized | 100% | 93.05% | 99.06% |

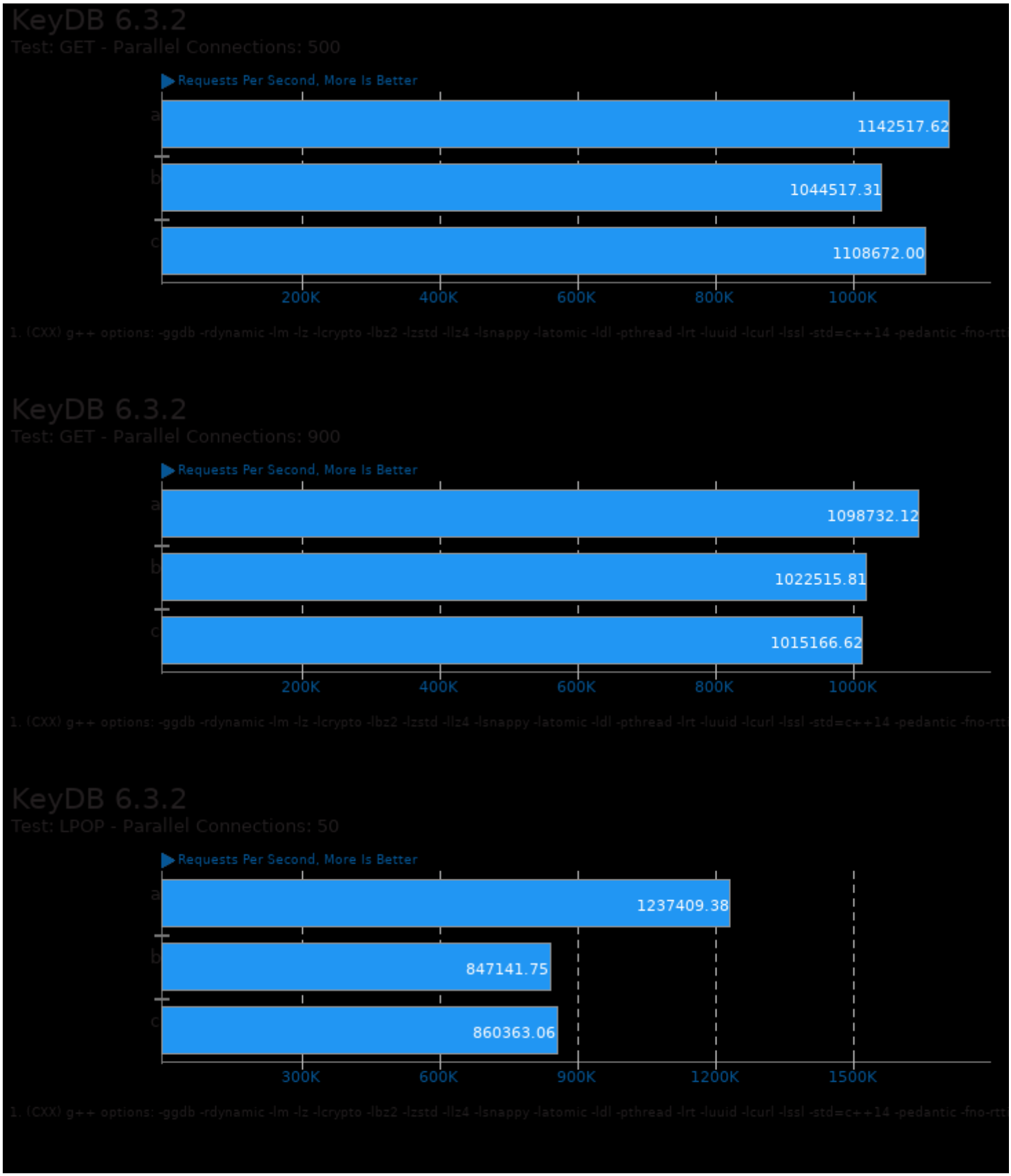
# ds key

| | | | |
|---|---|---|---|
| **Neural Magic DeepSparse - N.T.C.D.m - S.S.S (items/sec)** | 26.7169 | **26.8567** | **26.6154** |
| **Normalized** | 99.48% | 100% | 99.1% |
| **Neural Magic DeepSparse - N.T.C.D.m - S.S.S (ms/batch)** | 37.4214 | **37.2266** | **37.5642** |
| **Normalized** | 99.48% | 100% | 99.1% |
| **Neural Magic DeepSparse - C.S.9.P.Y.P - A.M.S (items/sec)** | **6.8672** | **6.8531** | 6.8583 |
| **Normalized** | 100% | 99.79% | 99.87% |
| **Neural Magic DeepSparse - C.S.9.P.Y.P - A.M.S (ms/batch)** | **291.2011** | **291.8002** | 291.575 |
| **Normalized** | 100% | 99.79% | 99.87% |
| **Neural Magic DeepSparse - C.S.9.P.Y.P - S.S.S (items/sec)** | 6.826 | **6.8287** | **6.8067** |
| **Normalized** | 99.96% | 100% | 99.68% |
| **Neural Magic DeepSparse - C.S.9.P.Y.P - S.S.S (ms/batch)** | 146.4805 | **146.4225** | **146.8947** |
| **Normalized** | 99.96% | 100% | 99.68% |
| **Neural Magic DeepSparse - N.T.C.B.b.u.S - A.M.S (items/sec)** | 13.9641 | **14.0144** | **13.9639** |
| **Normalized** | 99.64% | 100% | 99.64% |
| **Neural Magic DeepSparse - N.T.C.B.b.u.S - A.M.S (ms/batch)** | 143.1989 | **142.6845** | **143.2006** |
| **Normalized** | 99.64% | 100% | 99.64% |
| **Neural Magic DeepSparse - N.T.C.B.b.u.S - S.S.S (items/sec)** | **13.2023** | **13.3236** | 13.3 |
| **Normalized** | 99.09% | 100% | 99.82% |
| **Neural Magic DeepSparse - N.T.C.B.b.u.S - S.S.S (ms/batch)** | **75.7368** | **75.0465** | 75.1797 |
| **Normalized** | 99.09% | 100% | 99.82% |
| **Neural Magic DeepSparse - N.T.C.B.b.u.c - A.M.S (items/sec)** | **3.2707** | **3.1604** | 3.2108 |
| **Normalized** | 100% | 96.63% | 98.17% |
| **Neural Magic DeepSparse - N.T.C.B.b.u.c - A.M.S (ms/batch)** | **611.4515** | **632.7937** | 622.8627 |
| **Normalized** | 100% | 96.63% | 98.17% |
| **Neural Magic DeepSparse - N.T.C.B.b.u.c - S.S.S (items/sec)** | **3.2194** | **3.2087** | 3.2088 |
| **Normalized** | 100% | 99.67% | 99.67% |
| **Neural Magic DeepSparse - N.T.C.B.b.u.c - S.S.S (ms/batch)** | **310.6043** | **311.6435** | 311.6303 |
| **Normalized** | 100% | 99.67% | 99.67% |

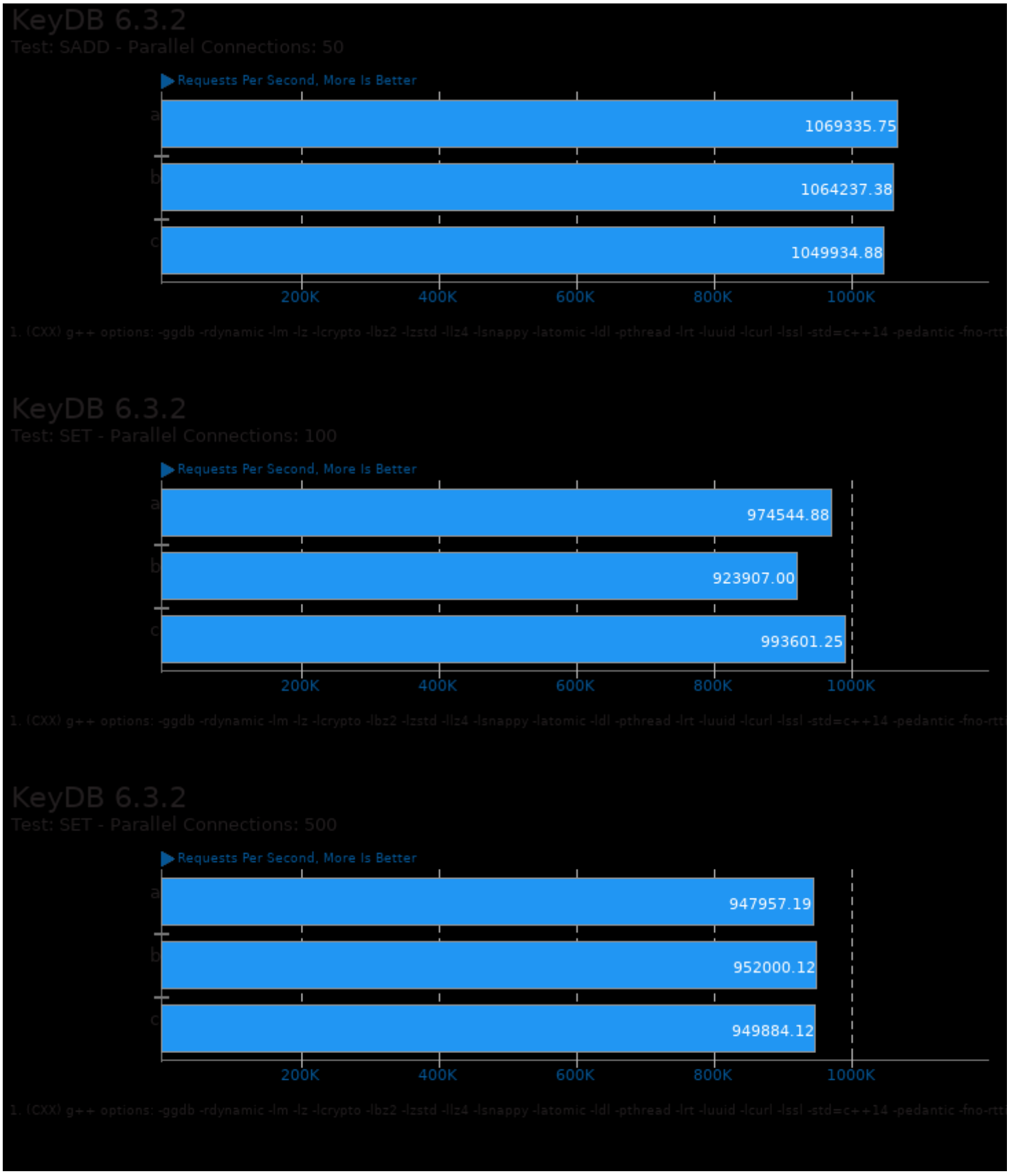
# ds key


KeyDB 6.3.2
Test: GET - Parallel Connections: 50
Requests Per Second, More Is Better
a
b
c
1177884.00
1135615.12
1108278.75
300K
600K
900K
1200K
1500K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtt
KeyDB 6.3.2
Test: SET - Parallel Connections: 50
Requests Per Second, More Is Better
a
b
c
1016156.88
1026609.75
970270.88
200K
400K
600K
800K
1000K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtt
KeyDB 6.3.2
Test: GET - Parallel Connections: 100
Requests Per Second, More Is Better
a
b
c
1149108.25
1102925.00
1092729.00
200K
400K
600K
800K
1000K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtt

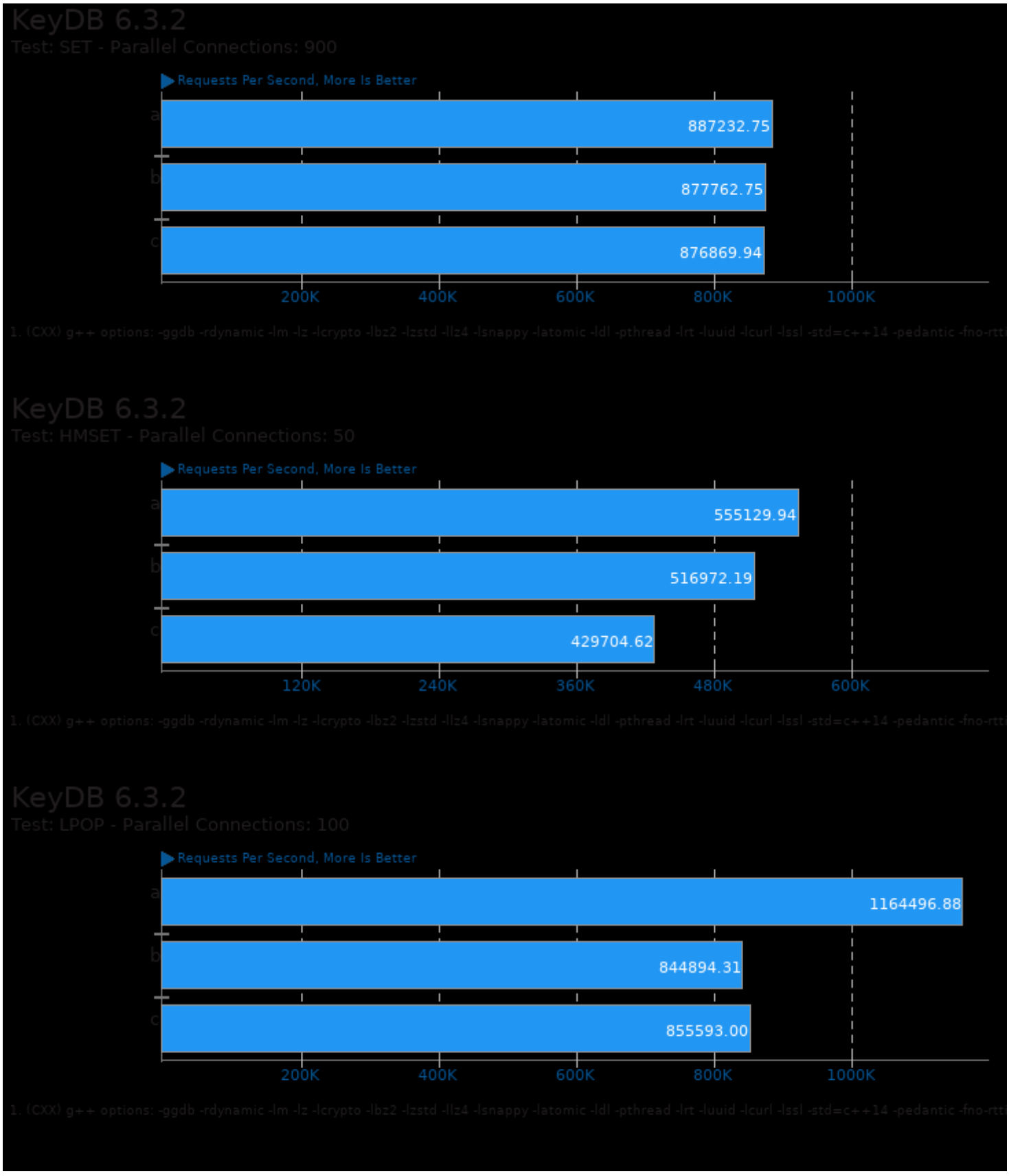
# ds key


KeyDB 6.3.2
Test: GET - Parallel Connections: 500
Requests Per Second, More Is Better
a
1142517.62
b
1044517.31
c
1108672.00
200K
400K
600K
800K
1000K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtt
KeyDB 6.3.2
Test: GET - Parallel Connections: 900
Requests Per Second, More Is Better
a
1098732.12
b
1022515.81
c
1015166.62
200K
400K
600K
800K
1000K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtt
KeyDB 6.3.2
Test: LPOP - Parallel Connections: 50
Requests Per Second, More Is Better
a
1237409.38
b
847141.75
c
860363.06
300K
600K
900K
1200K
1500K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtt

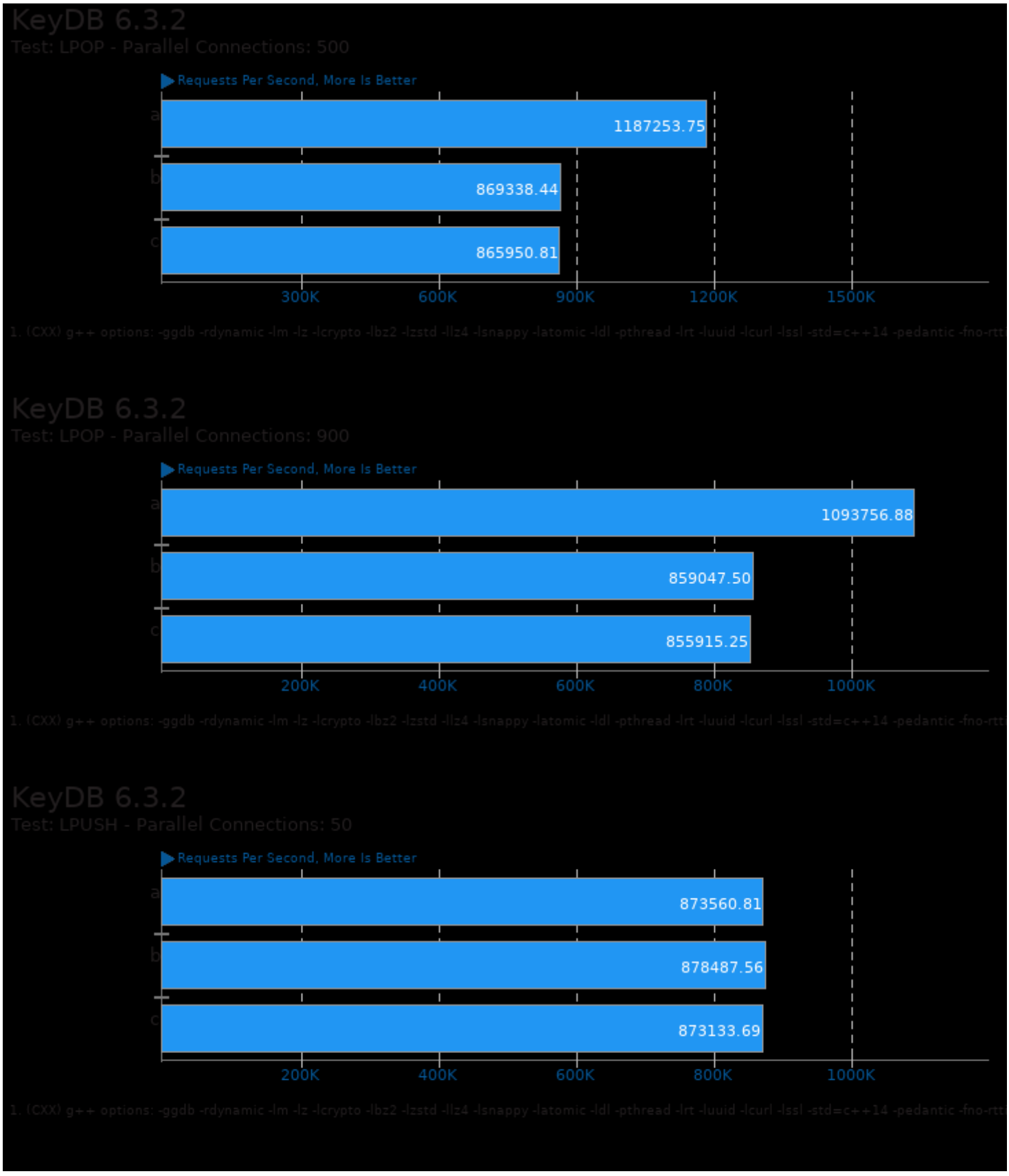
# ds key


KeyDB 6.3.2
Test: SADD - Parallel Connections: 50
Requests Per Second, More Is Better
a
b
c
1069335.75
1064237.38
1049934.88
200K
400K
600K
800K
1000K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtt
KeyDB 6.3.2
Test: SET - Parallel Connections: 100
Requests Per Second, More Is Better
a
b
c
974544.88
923907.00
993601.25
200K
400K
600K
800K
1000K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtt
KeyDB 6.3.2
Test: SET - Parallel Connections: 500
Requests Per Second, More Is Better
a
b
c
947957.19
952000.12
949884.12
200K
400K
600K
800K
1000K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtt

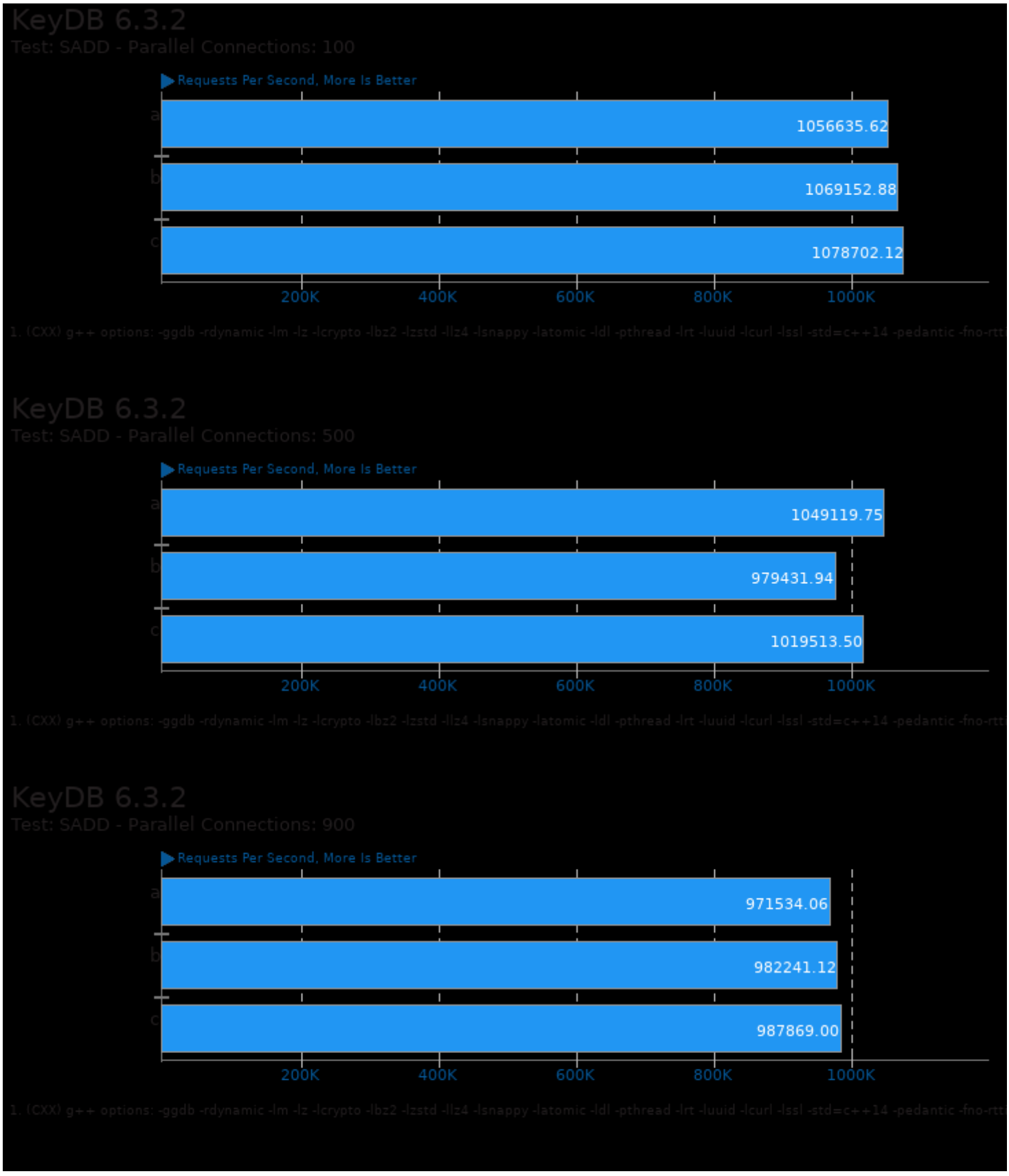
# ds key


KeyDB 6.3.2
Test: SET - Parallel Connections: 900
Requests Per Second, More Is Better
a
887232.75
b
877762.75
c
876869.94
200K
400K
600K
800K
1000K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtt
KeyDB 6.3.2
Test: HMSET - Parallel Connections: 50
Requests Per Second, More Is Better
a
555129.94
b
516972.19
c
429704.62
120K
240K
360K
480K
600K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtt
KeyDB 6.3.2
Test: LPOP - Parallel Connections: 100
Requests Per Second, More Is Better
a
1164496.88
b
844894.31
c
855593.00
200K
400K
600K
800K
1000K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtt

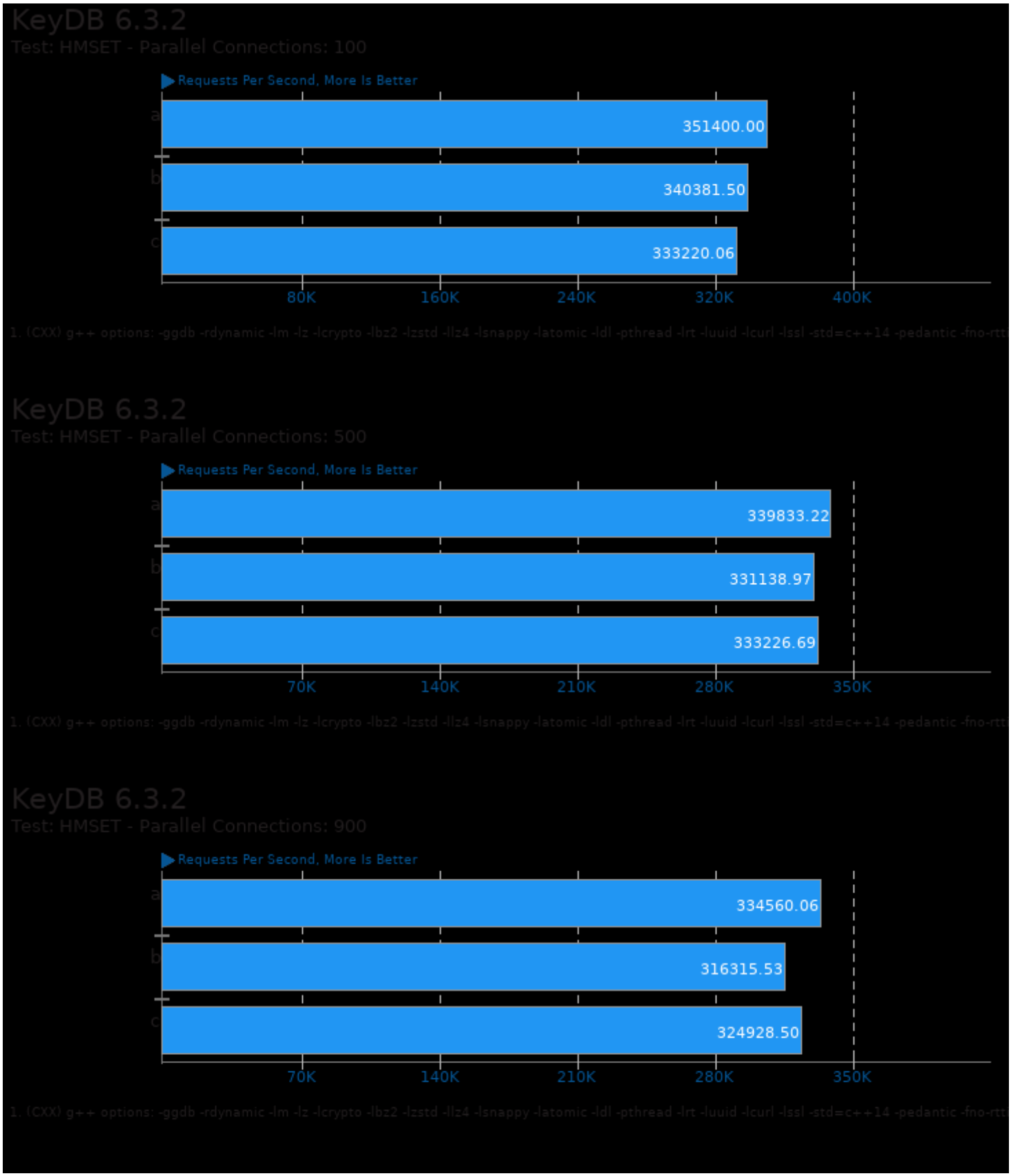
# ds key


KeyDB 6.3.2
Test: LPOP - Parallel Connections: 500
Requests Per Second, More Is Better
a
1187253.75
b
869338.44
c
865950.81
300K
600K
900K
1200K
1500K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtt
KeyDB 6.3.2
Test: LPOP - Parallel Connections: 900
Requests Per Second, More Is Better
a
1093756.88
b
859047.50
c
855915.25
200K
400K
600K
800K
1000K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtt
KeyDB 6.3.2
Test: LPUSH - Parallel Connections: 50
Requests Per Second, More Is Better
a
873560.81
b
878487.56
c
873133.69
200K
400K
600K
800K
1000K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtt

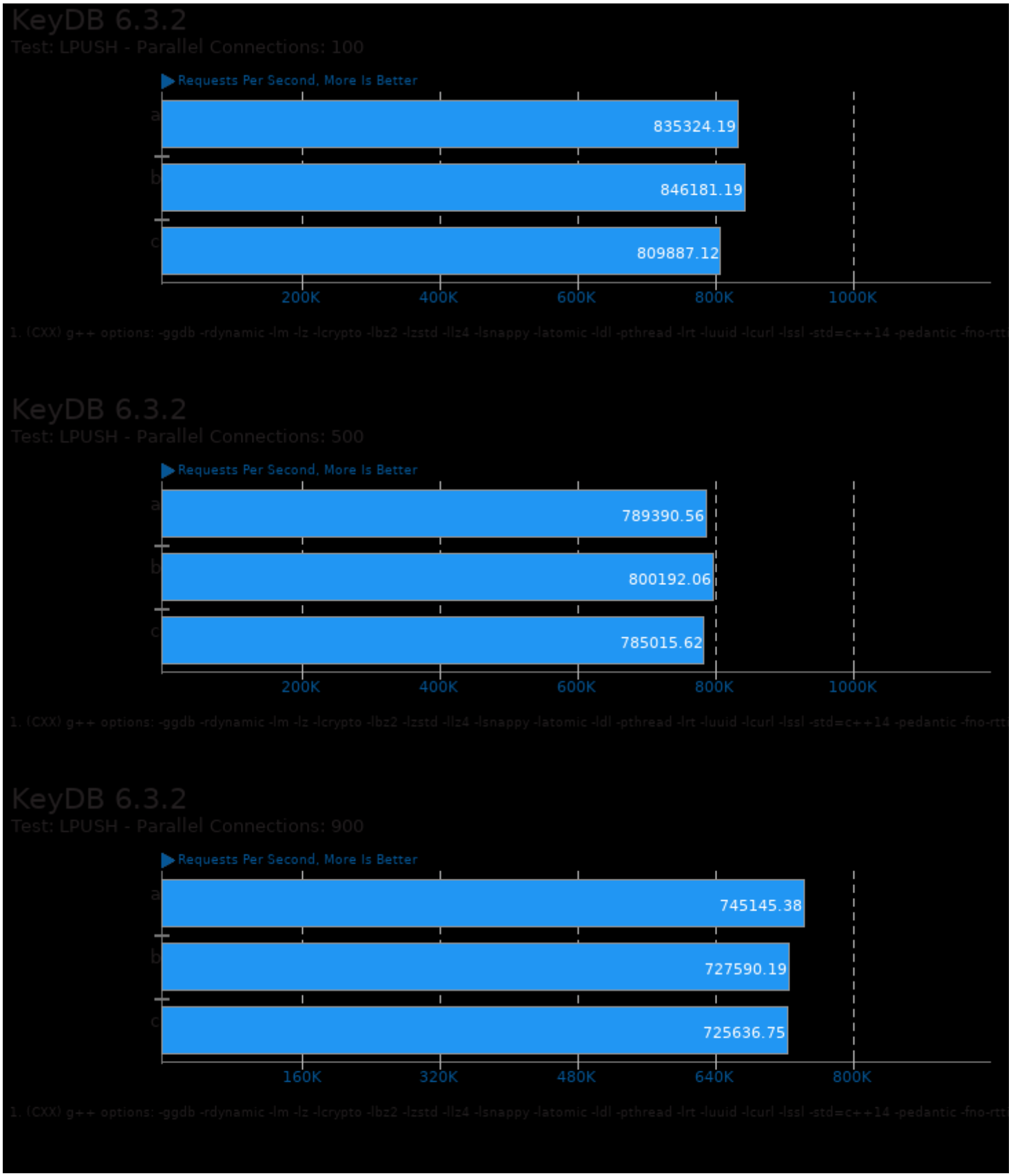
# ds key


KeyDB 6.3.2
Test: SADD - Parallel Connections: 100
Requests Per Second, More Is Better
a
1056635.62
b
1069152.88
c
1078702.12
200K
400K
600K
800K
1000K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtt
KeyDB 6.3.2
Test: SADD - Parallel Connections: 500
Requests Per Second, More Is Better
a
1049119.75
b
979431.94
c
1019513.50
200K
400K
600K
800K
1000K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtt
KeyDB 6.3.2
Test: SADD - Parallel Connections: 900
Requests Per Second, More Is Better
a
971534.06
b
982241.12
c
987869.00
200K
400K
600K
800K
1000K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtt

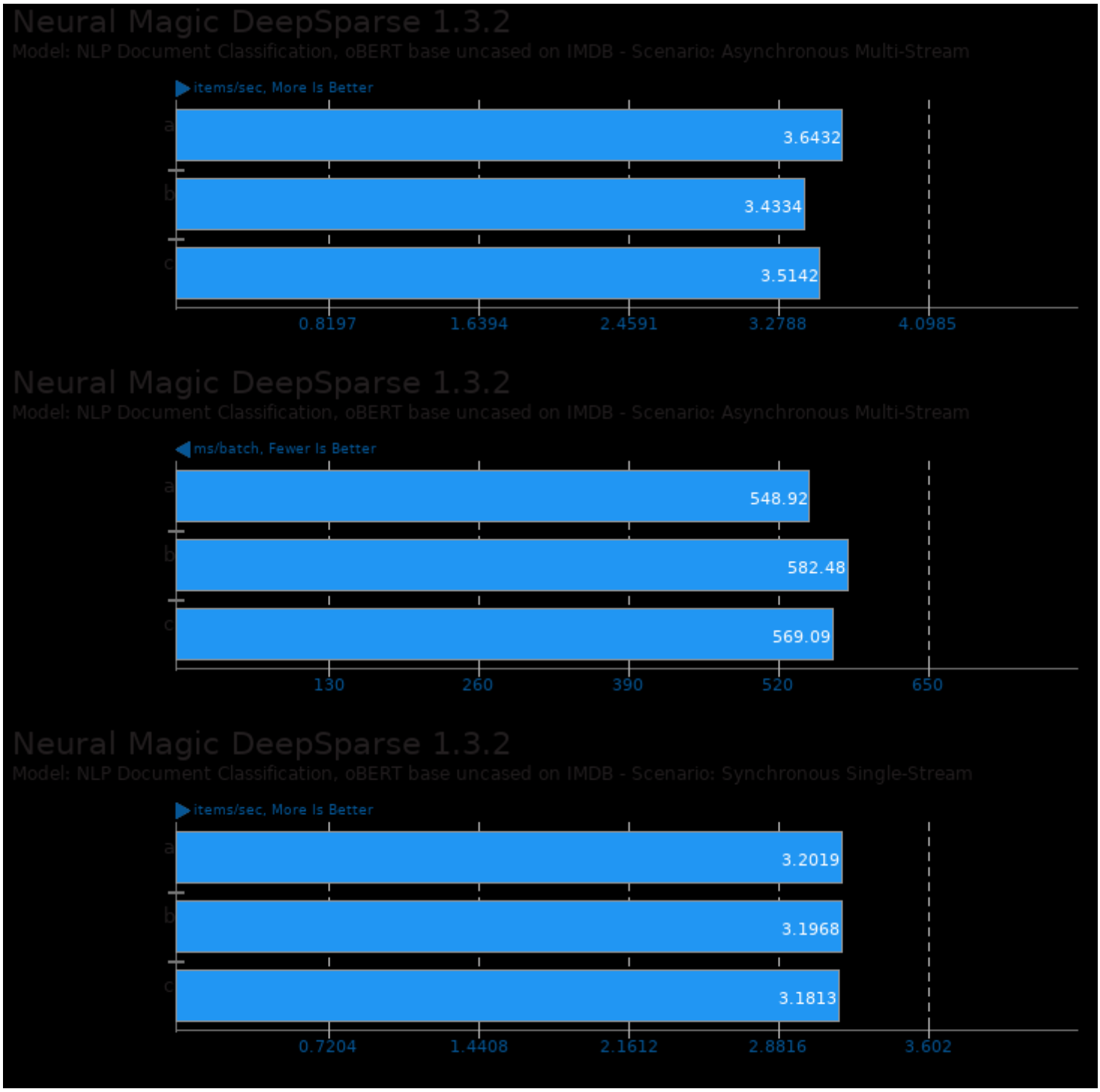
# ds key


KeyDB 6.3.2
Test: HMSET - Parallel Connections: 100
Requests Per Second, More Is Better
a
b
c
351400.00
340381.50
333220.06
80K
160K
240K
320K
400K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtt
KeyDB 6.3.2
Test: HMSET - Parallel Connections: 500
Requests Per Second, More Is Better
a
b
c
339833.22
331138.97
333226.69
70K
140K
210K
280K
350K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtt
KeyDB 6.3.2
Test: HMSET - Parallel Connections: 900
Requests Per Second, More Is Better
a
b
c
334560.06
316315.53
324928.50
70K
140K
210K
280K
350K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtt

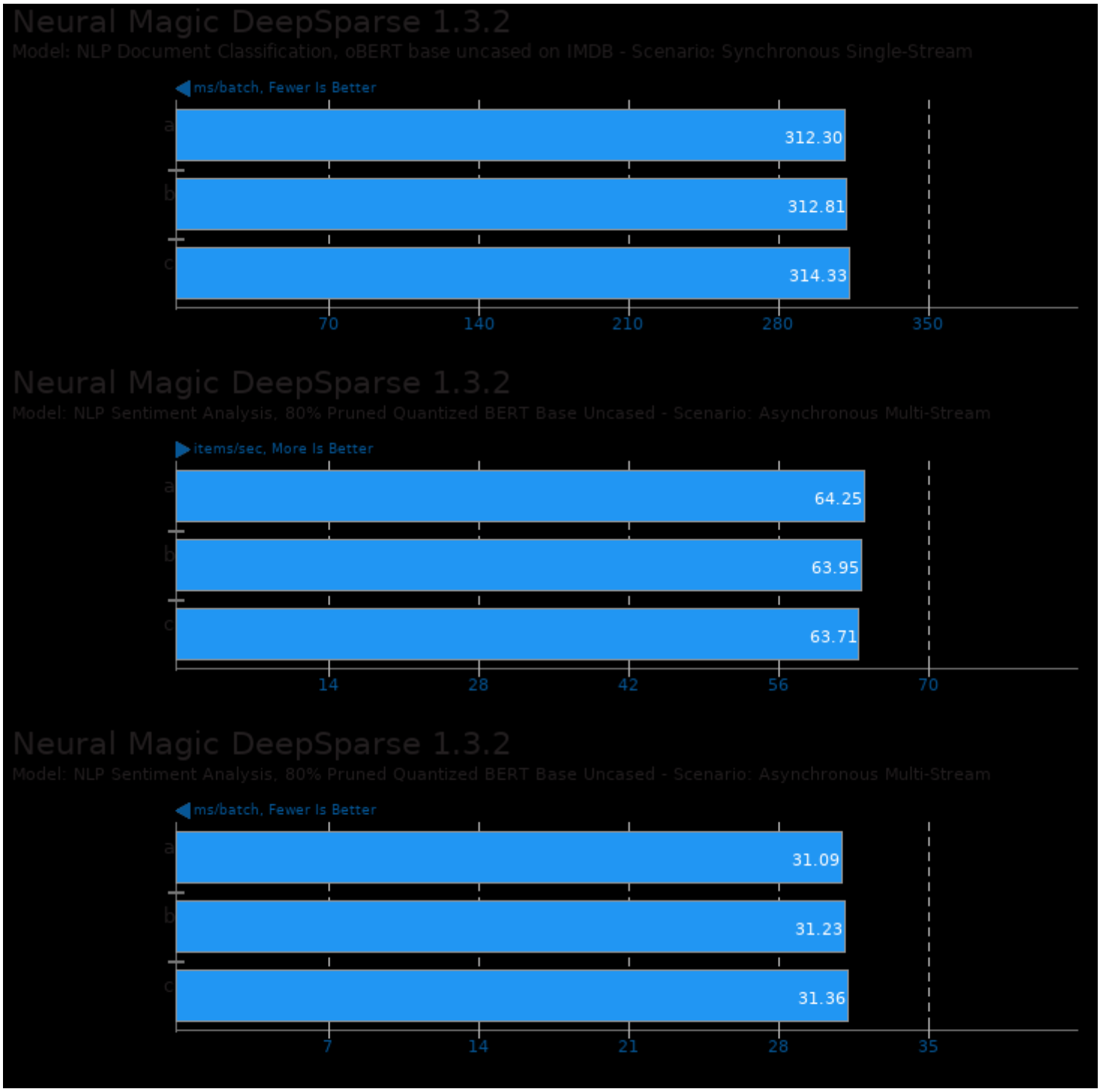
# ds key


KeyDB 6.3.2
Test: LPUSH - Parallel Connections: 100
Requests Per Second, More Is Better
a
835324.19
b
846181.19
c
809887.12
200K
400K
600K
800K
1000K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtt
KeyDB 6.3.2
Test: LPUSH - Parallel Connections: 500
Requests Per Second, More Is Better
a
789390.56
b
800192.06
c
785015.62
200K
400K
600K
800K
1000K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtt
KeyDB 6.3.2
Test: LPUSH - Parallel Connections: 900
Requests Per Second, More Is Better
a
745145.38
b
727590.19
c
725636.75
160K
320K
480K
640K
800K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtt

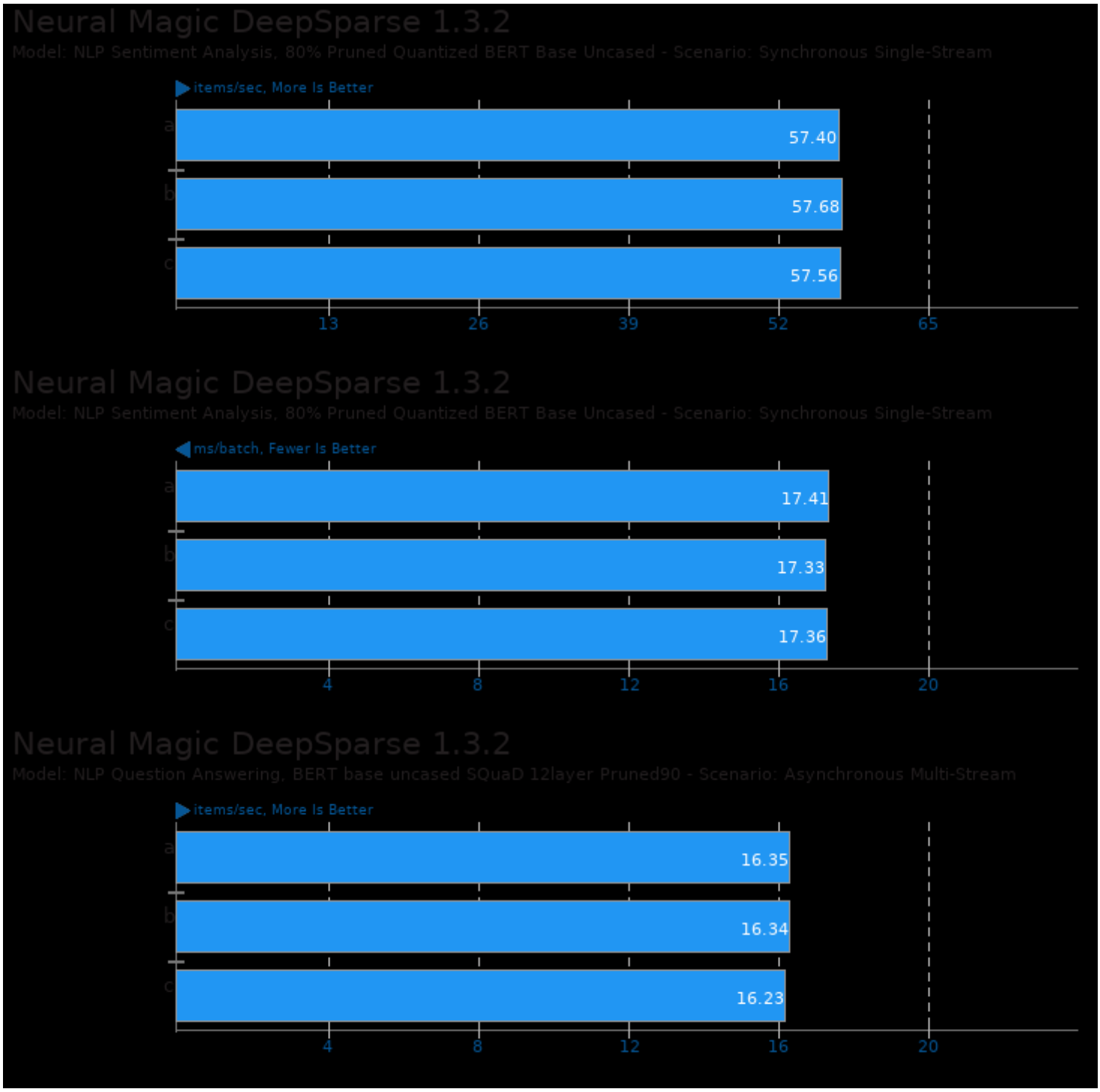
# ds key


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
3.6432
3.4334
3.5142
0.8197
1.6394
2.4591
3.2788
4.0985
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
548.92
582.48
569.09
130
260
390
520
650
Neural Magic DeepSparse 1.3.2
Model: NLP Document Classification, oBERT base uncased on IMDB - Scenario: Synchronous Single-Stream
items/sec, More Is Better
a
b
c
3.2019
3.1968
3.1813
0.7204
1.4408
2.1612
2.8816
3.602

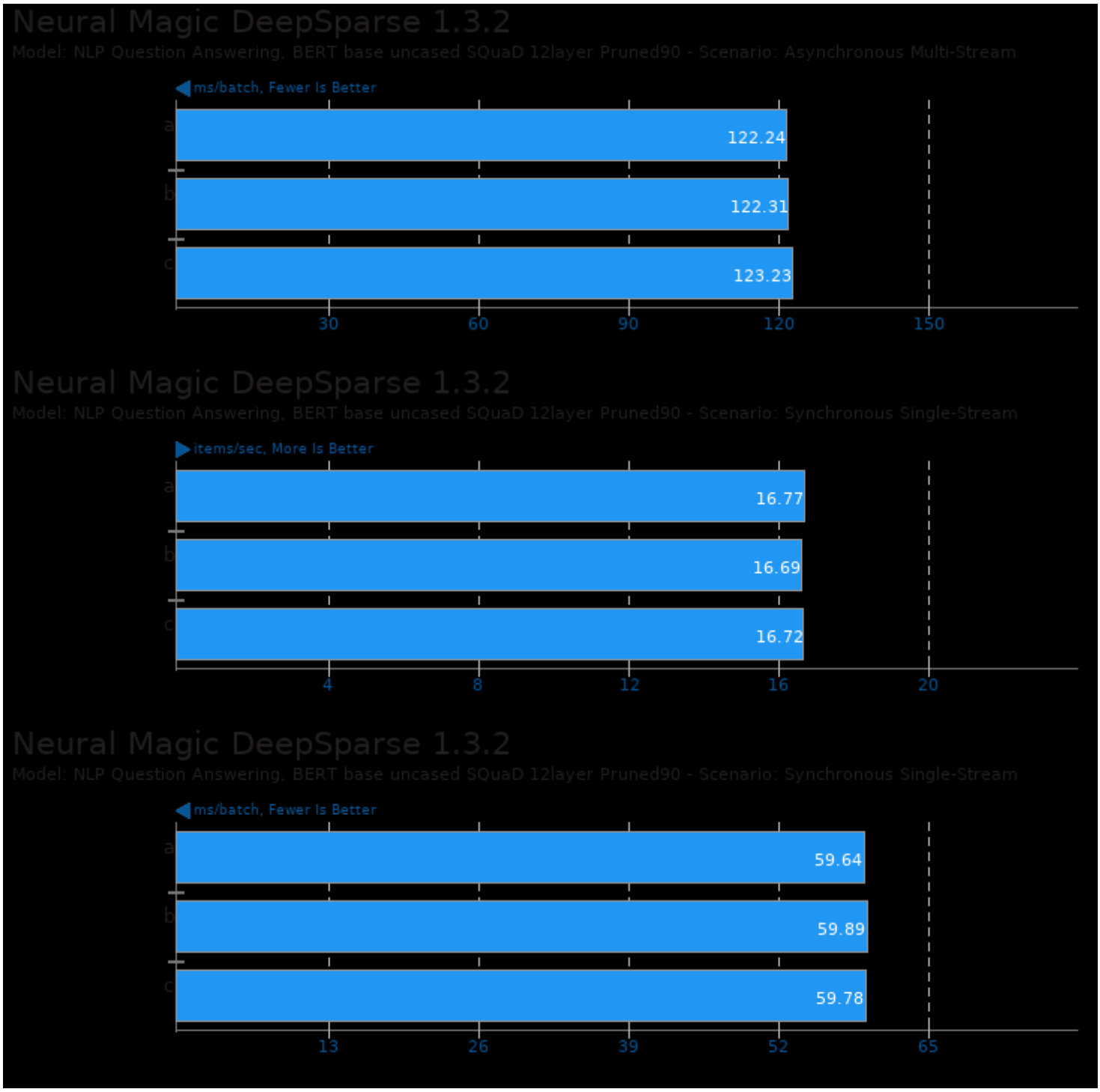
# ds key


Neural Magic DeepSparse 1.3.2
Model: NLP Document Classification, oBERT base uncased on IMDB - Scenario: Synchronous Single-Stream
ms/batch, Fewer Is Better
a
b
c
312.30
312.81
314.33
70
140
210
280
350
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
64.25
63.95
63.71
14
28
42
56
70
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
31.09
31.23
31.36
7
14
21
28
35

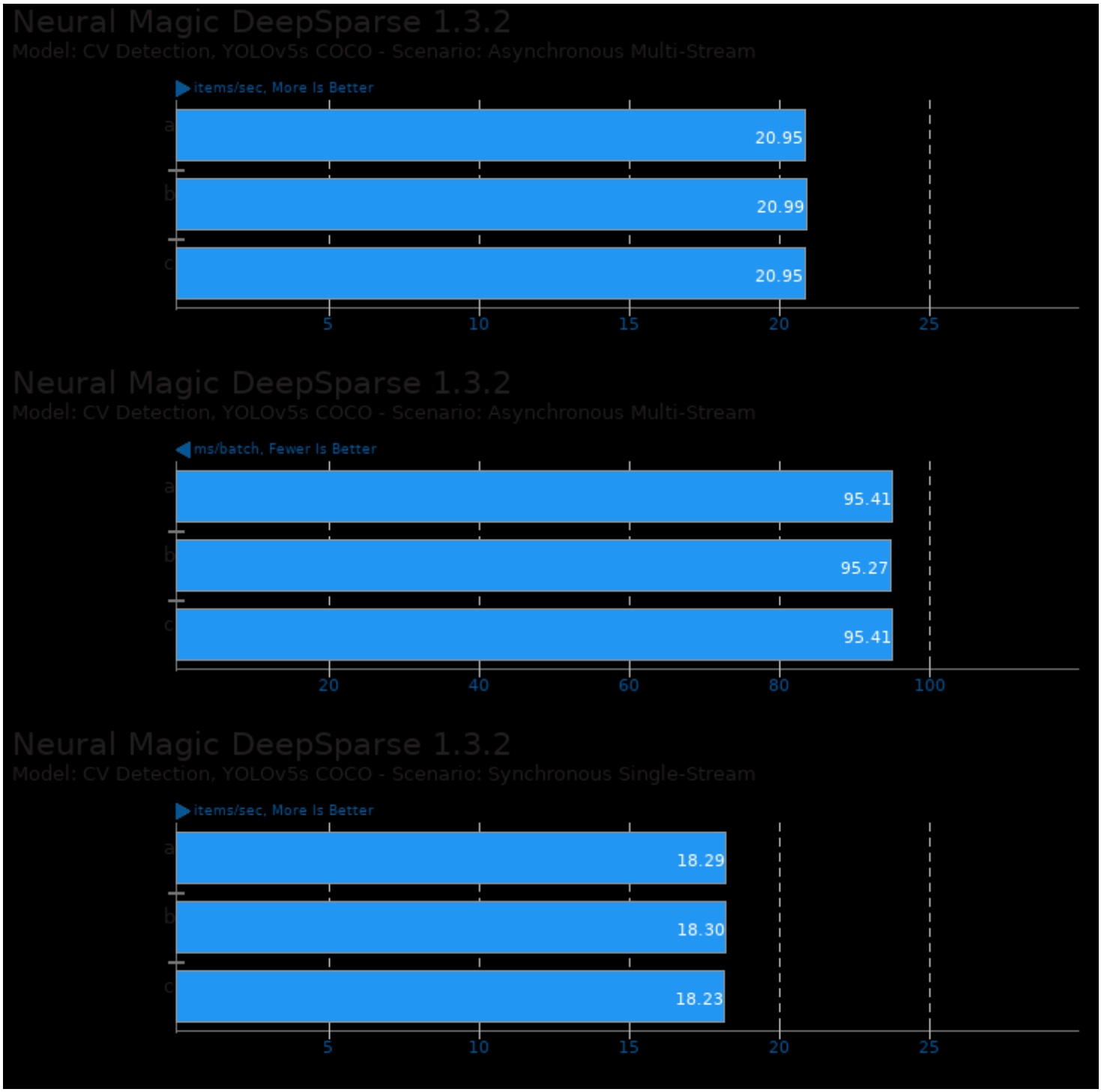
# ds key


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
57.40
57.68
57.56
13
26
39
52
65
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
17.41
17.33
17.36
4
8
12
16
20
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
16.35
16.34
16.23
4
8
12
16
20

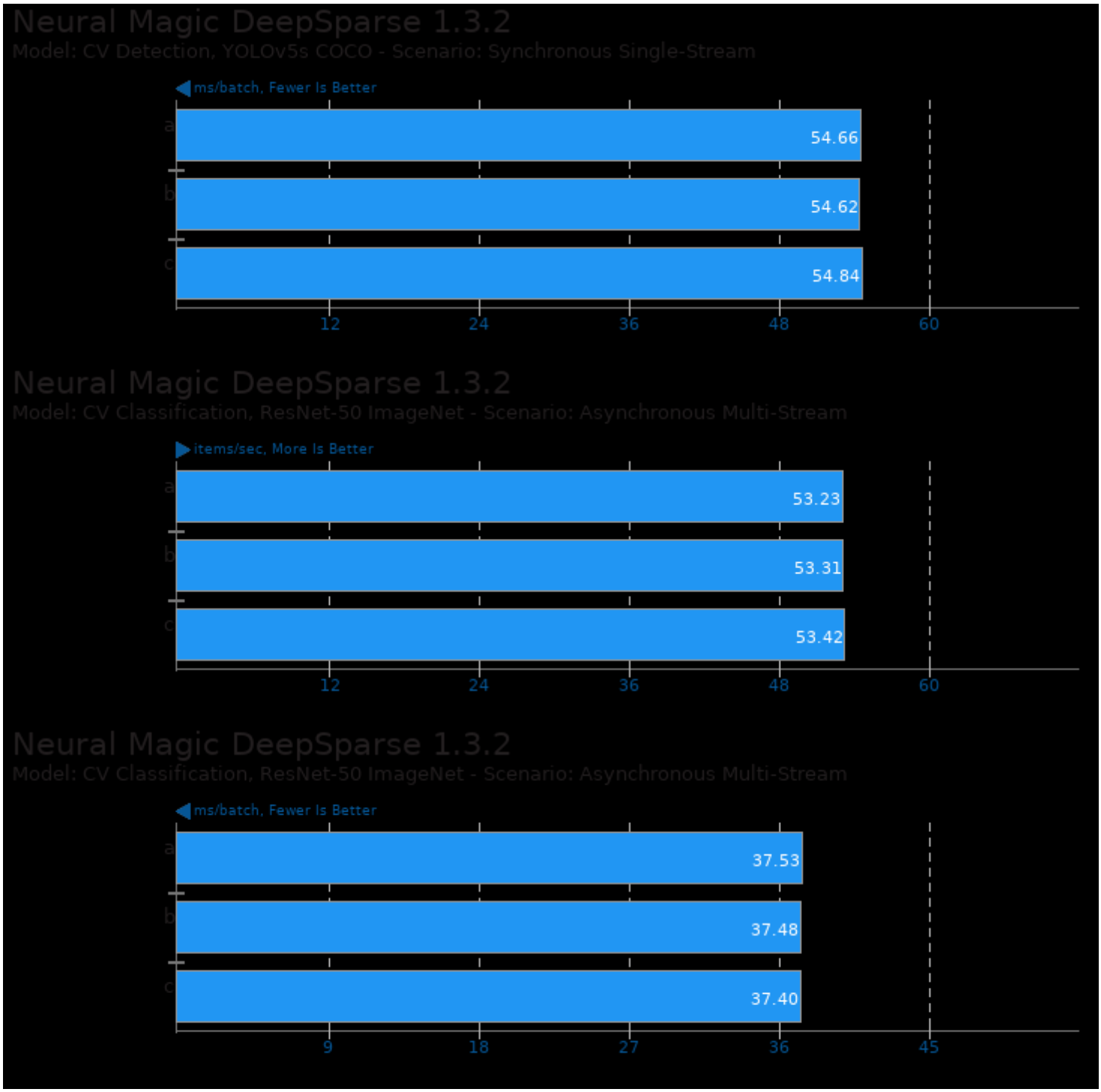
# ds key


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
122.24
122.31
123.23
30
60
90
120
150
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
16.77
16.69
16.72
4
8
12
16
20
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
59.64
59.89
59.78
13
26
39
52
65

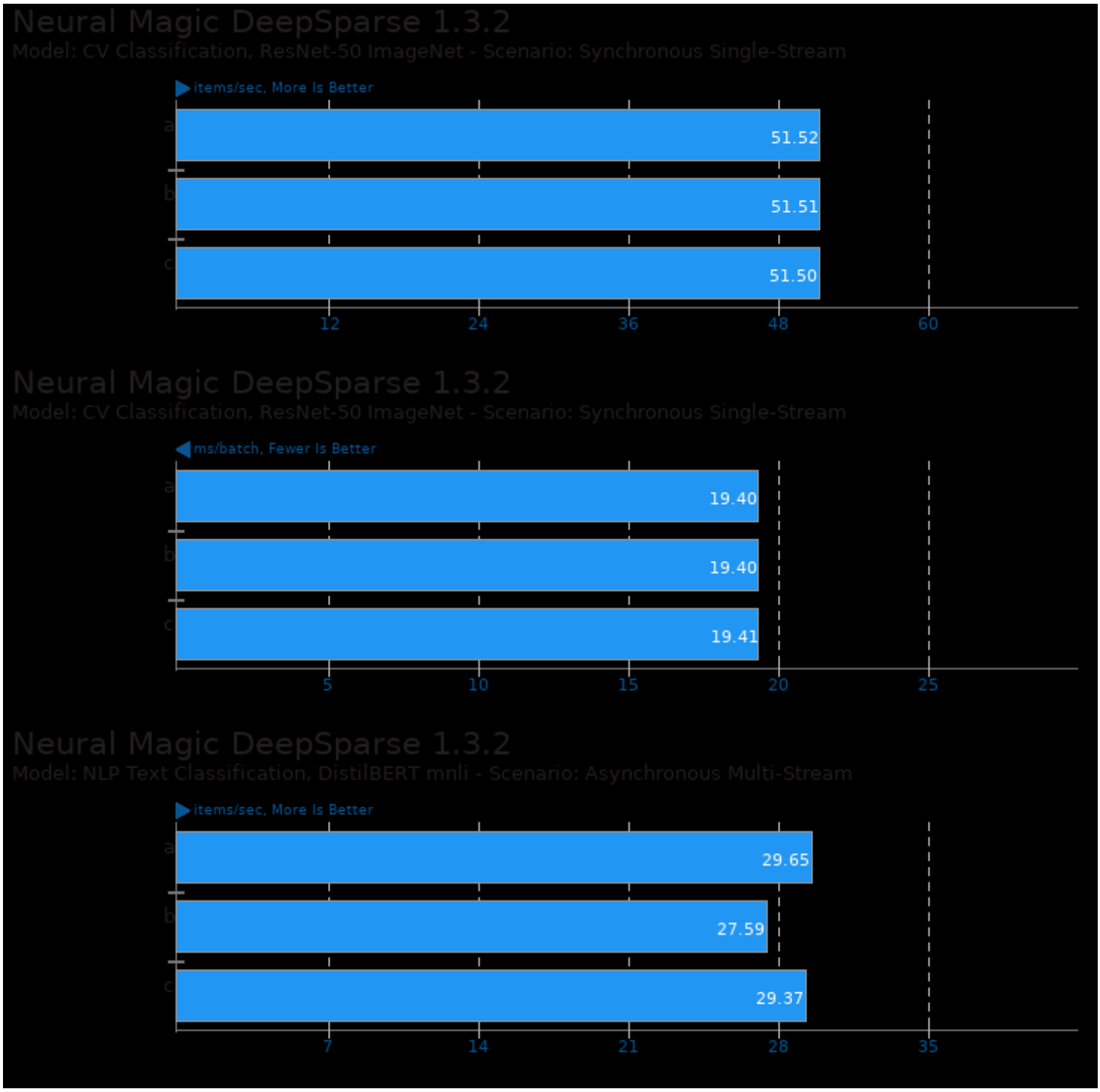
# ds key


Neural Magic DeepSparse 1.3.2
Model: CV Detection, YOLOv5s COCO - Scenario: Asynchronous Multi-Stream
items/sec, More Is Better
a
b
c
20.95
20.99
20.95
5
10
15
20
25
Neural Magic DeepSparse 1.3.2
Model: CV Detection, YOLOv5s COCO - Scenario: Asynchronous Multi-Stream
ms/batch, Fewer Is Better
a
b
c
95.41
95.27
95.41
20
40
60
80
100
Neural Magic DeepSparse 1.3.2
Model: CV Detection, YOLOv5s COCO - Scenario: Synchronous Single-Stream
items/sec, More Is Better
a
b
c
18.29
18.30
18.23
5
10
15
20
25

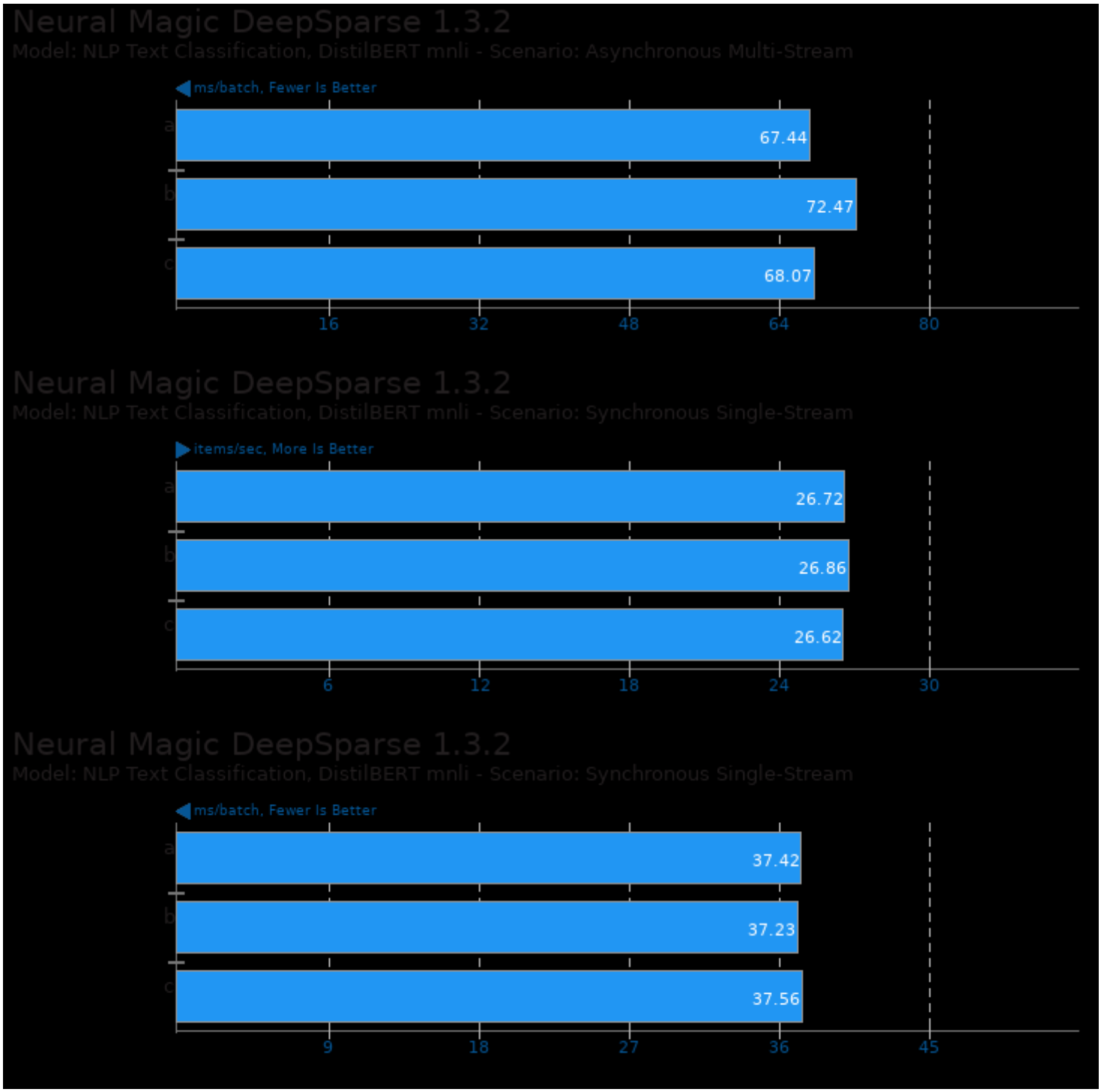
# ds key

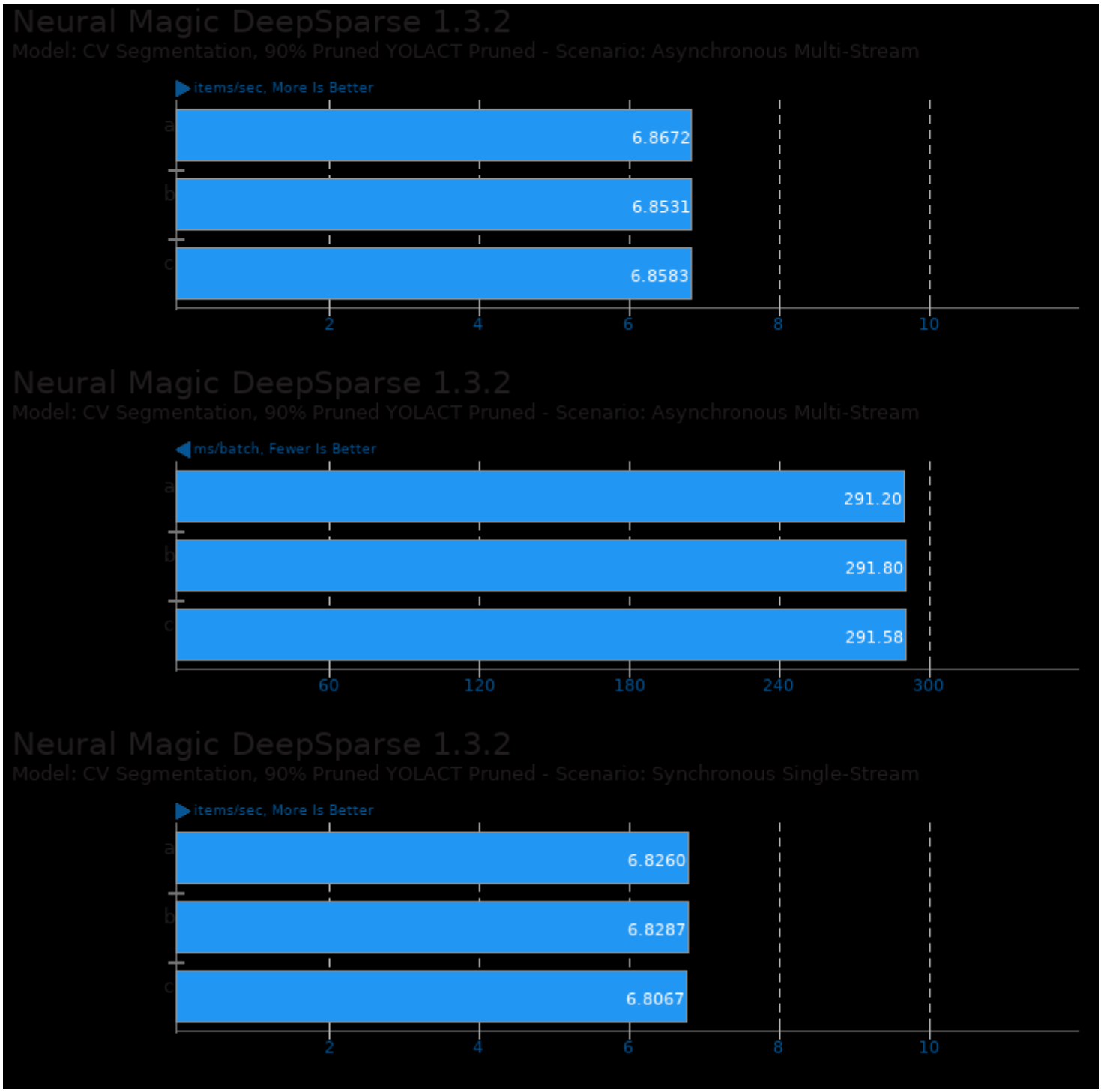

Neural Magic DeepSparse 1.3.2
Model: CV Detection, YOLOv5s COCO - Scenario: Synchronous Single-Stream
ms/batch, Fewer Is Better
a
b
c
54.66
54.62
54.84
12
24
36
48
60
Neural Magic DeepSparse 1.3.2
Model: CV Classification, ResNet-50 ImageNet - Scenario: Asynchronous Multi-Stream
items/sec, More Is Better
a
b
c
53.23
53.31
53.42
12
24
36
48
60
Neural Magic DeepSparse 1.3.2
Model: CV Classification, ResNet-50 ImageNet - Scenario: Asynchronous Multi-Stream
ms/batch, Fewer Is Better
a
b
c
37.53
37.48
37.40
9
18
27
36
45

# ds key

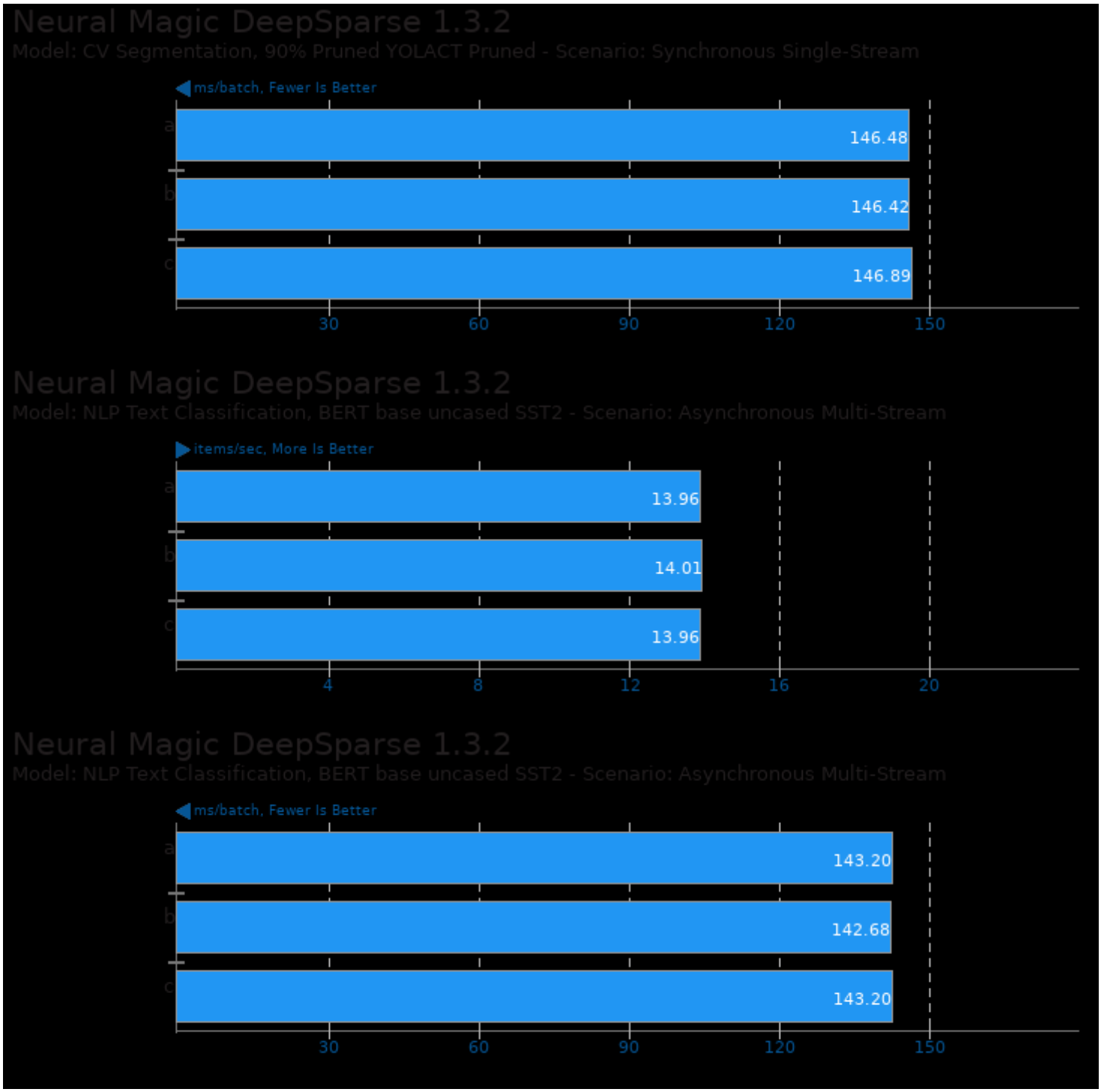

Neural Magic DeepSparse 1.3.2
Model: CV Classification, ResNet-50 ImageNet - Scenario: Synchronous Single-Stream
items/sec, More Is Better
a
b
c
51.52
51.51
51.50
12
24
36
48
60
Neural Magic DeepSparse 1.3.2
Model: CV Classification, ResNet-50 ImageNet - Scenario: Synchronous Single-Stream
ms/batch, Fewer Is Better
a
b
c
19.40
19.40
19.41
5
10
15
20
25
Neural Magic DeepSparse 1.3.2
Model: NLP Text Classification, DistilBERT mnli - Scenario: Asynchronous Multi-Stream
items/sec, More Is Better
a
b
c
29.65
27.59
29.37
7
14
21
28
35

# ds key

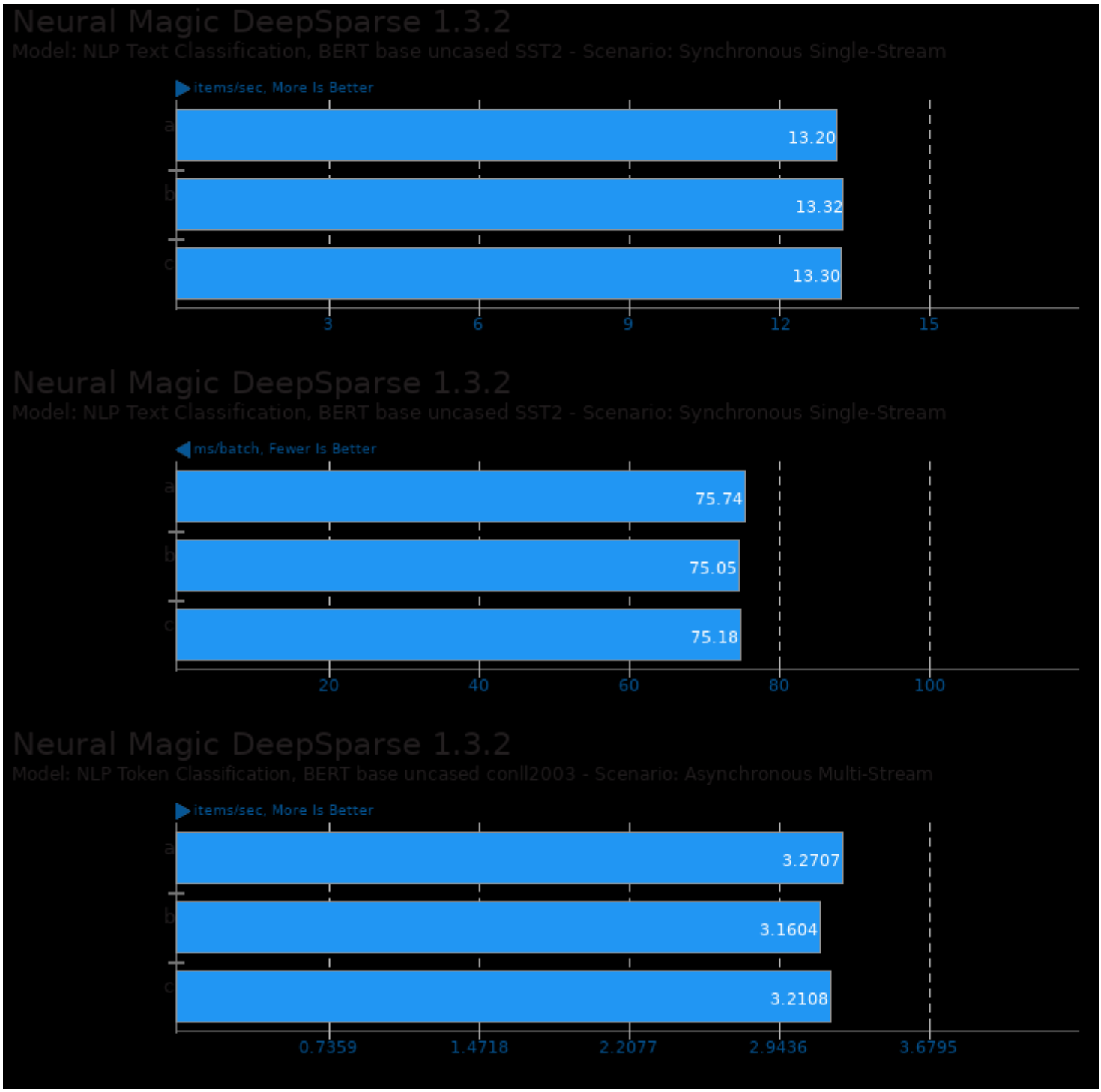

Neural Magic DeepSparse 1.3.2
Model: NLP Text Classification, DistilBERT mnli - Scenario: Asynchronous Multi-Stream
ms/batch, Fewer Is Better
a
b
c
67.44
72.47
68.07
16
32
48
64
80
Neural Magic DeepSparse 1.3.2
Model: NLP Text Classification, DistilBERT mnli - Scenario: Synchronous Single-Stream
items/sec, More Is Better
a
b
c
26.72
26.86
26.62
6
12
18
24
30
Neural Magic DeepSparse 1.3.2
Model: NLP Text Classification, DistilBERT mnli - Scenario: Synchronous Single-Stream
ms/batch, Fewer Is Better
a
b
c
37.42
37.23
37.56
9
18
27
36
45

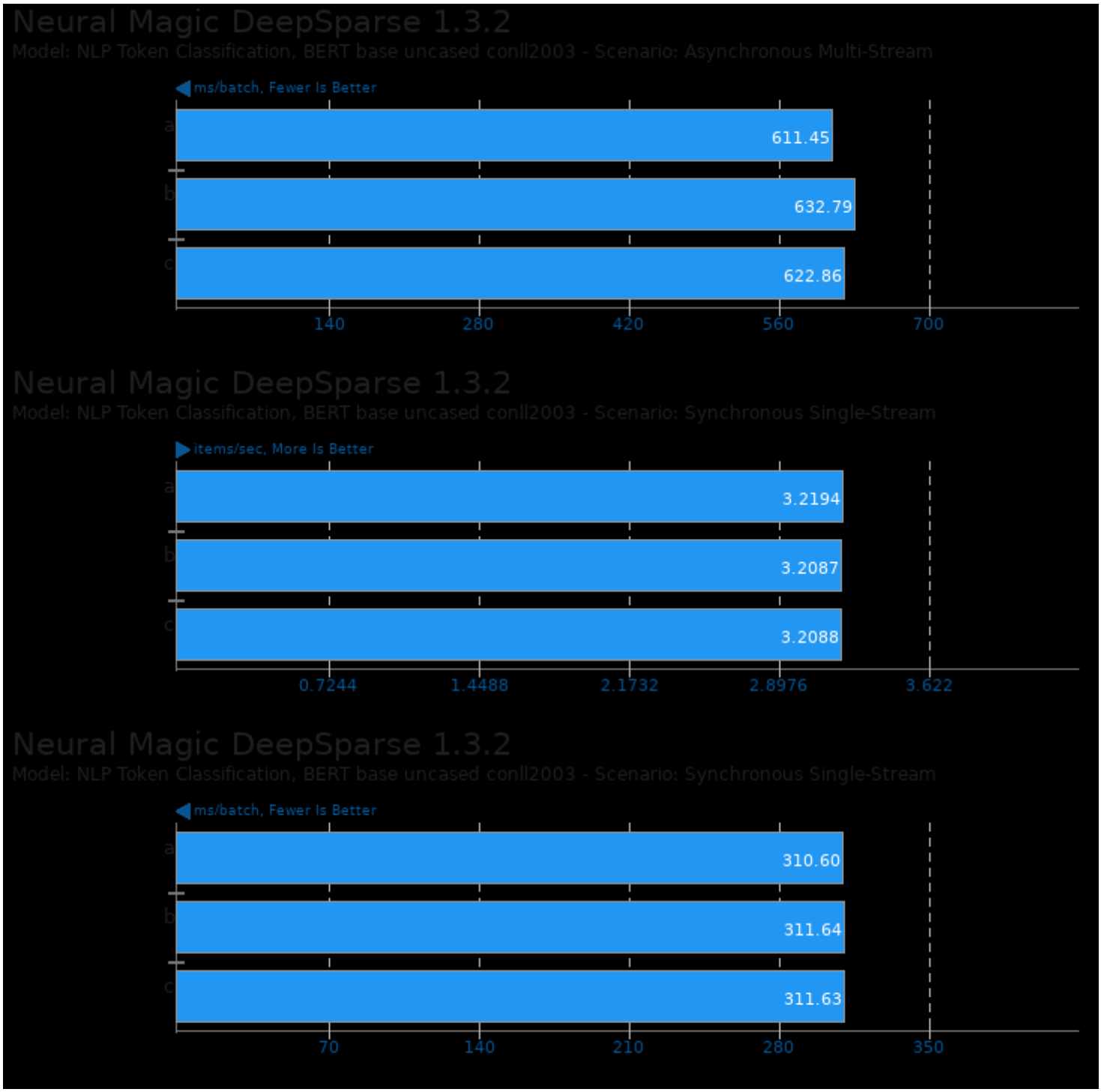

Neural Magic DeepSparse 1.3.2
Model: CV Segmentation, 90% Pruned YOLACT Pruned - Scenario: Asynchronous Multi-Stream
items/sec, More Is Better
a
b
c
6.8672
6.8531
6.8583
2
4
6
8
10
Neural Magic DeepSparse 1.3.2
Model: CV Segmentation, 90% Pruned YOLACT Pruned - Scenario: Asynchronous Multi-Stream
ms/batch, Fewer Is Better
a
b
c
291.20
291.80
291.58
60
120
180
240
300
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
6.8260
6.8287
6.8067
2
4
6
8
10

# ds key


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
146.48
146.42
146.89
30
60
90
120
150
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
13.96
14.01
13.96
4
8
12
16
20
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
143.20
142.68
143.20
30
60
90
120
150

# ds key


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
13.20
13.32
13.30
3
6
9
12
15
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
75.74
75.05
75.18
20
40
60
80
100
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
3.2707
3.1604
3.2108
0.7359
1.4718
2.2077
2.9436
3.6795




*This file was automatically generated via the Phoronix Test Suite benchmarking software on Friday, 29 March 2024 11:32.*