

www.phoronix-test-suite.com

icelake deep

Intel Core i7-1065G7 testing with a Dell 06CDVY (1.0.9 BIOS) and Intel Iris Plus ICL GT2 16GB 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 1.038x the speed of the slowest (b). c was 0.964x the speed of a and b was 1x the speed of c.

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

KeyDB (Test: SET - Parallel Connections: 500) at 1.673x
KeyDB (Test: LPOP - Parallel Connections: 50) at 1.601x
KeyDB (Test: LPOP - Parallel Connections: 500) at 1.52x
KeyDB (Test: LPOP - Parallel Connections: 100) at 1.506x
KeyDB (Test: LPOP - Parallel Connections: 900) at 1.26x
KeyDB (Test: GET - Parallel Connections: 900) at 1.237x
KeyDB (Test: SET - Parallel Connections: 100) at 1.203x
KeyDB (Test: GET - Parallel Connections: 500) at 1.198x
KeyDB (Test: GET - Parallel Connections: 100) at 1.183x

KeyDB (Test: SADD - Parallel Connections: 50) at 1.17x.

Test Systems:

a

b

c

Processor: Intel Core i7-1065G7 @ 3.90GHz (4 Cores / 8 Threads), Motherboard: Dell 06CDVY (1.0.9 BIOS), Chipset: Intel Ice Lake-LP DRAM, Memory: 16GB, Disk: Toshiba KBG40ZPZ512G NVMe 512GB, Graphics: Intel Iris Plus ICL GT2 16GB (1100MHz), Audio: Realtek ALC289, Network: Intel Ice Lake-LP PCH CNVi WiFi

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: 0xb2 - Thermald 2.4.9
Python Notes: Python 3.10.6
Security Notes: itlb_multihit: KVM: Mitigation of VMX disabled + l1tf: Not affected + mds: Not affected + meltdown: Not affected + mmio_stale_data: Mitigation of Clear buffers; SMT vulnerable + 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: Mitigation of Microcode + tsx_async_abort: Not affected

	a	b	c
Neural Magic DeepSparse - N.D.C.o.b.u.o.l - A.M.S (items/sec)	**1.9804**	1.9855	**1.9956**
Normalized	99.24%	99.49%	100%
Neural Magic DeepSparse - N.D.C.o.b.u.o.l - S.S.S (items/sec)	**1.9891**	**1.9858**	1.9861
Normalized	100%	99.83%	99.85%
Neural Magic DeepSparse - N.S.A.8.P.Q.B.B.U - A.M.S (items/sec)	**43.7176**	**42.468**	43.4887
Normalized	100%	97.14%	99.48%
Neural Magic DeepSparse - N.S.A.8.P.Q.B.B.U - S.S.S (items/sec)	**41.6174**	41.4098	**41.3882**
Normalized	100%	99.5%	99.45%

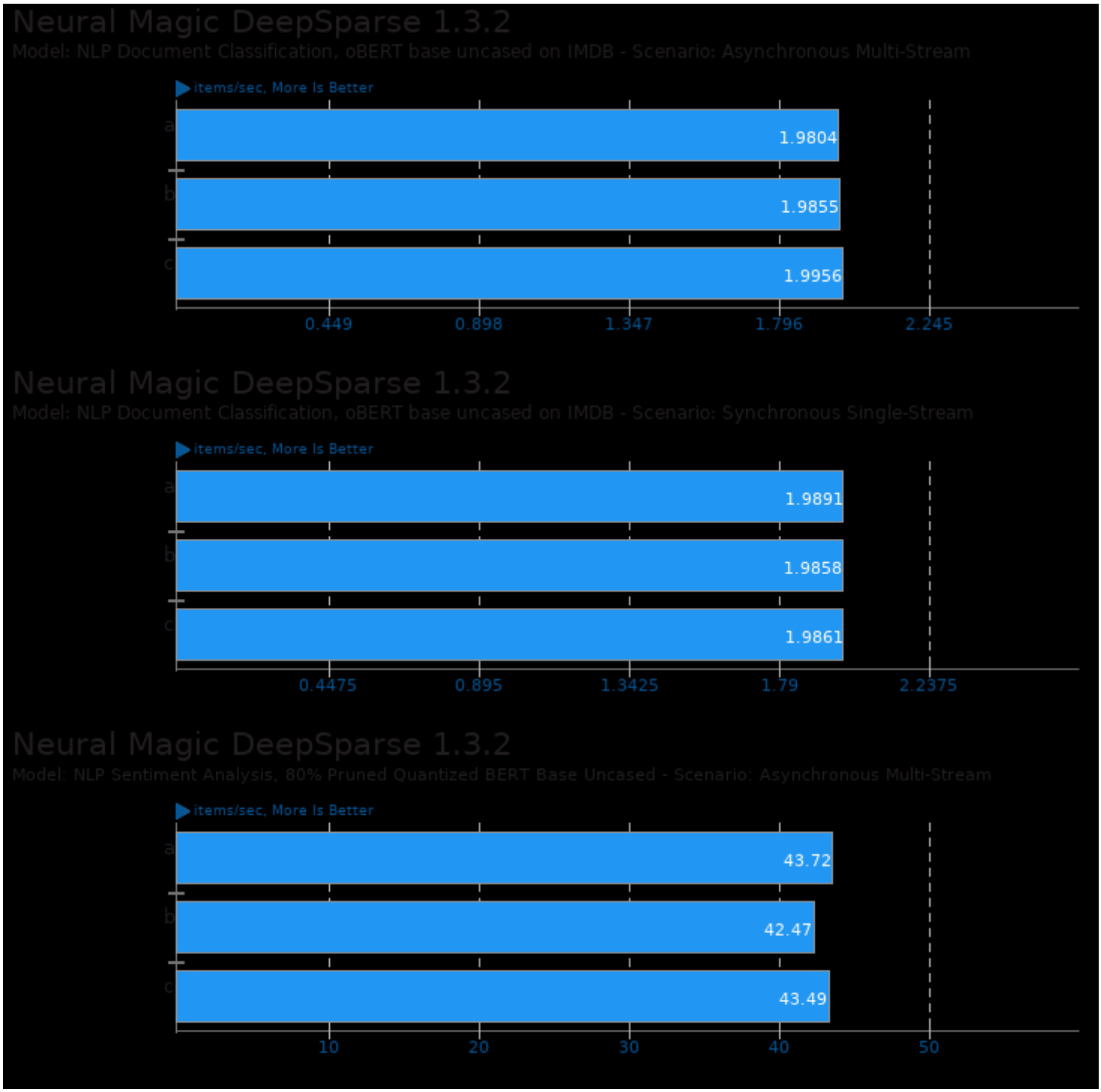
Neural Magic DeepSparse - N.Q.A.B.b.u.S.1.P - A.M.S (items/sec)	10.7539	10.7429	10.7395
Normalized	100%	99.9%	99.87%
Neural Magic DeepSparse - N.Q.A.B.b.u.S.1.P - S.S.S (items/sec)	10.7315	10.75	10.8242
Normalized	99.14%	99.31%	100%
Neural Magic DeepSparse - C.D.Y.C - A.M.S (items/sec)	13.6817	13.6358	13.5651
Normalized	100%	99.66%	99.15%
Neural Magic DeepSparse - C.D.Y.C - S.S.S (items/sec)	12.3534	12.3371	12.3143
Normalized	100%	99.87%	99.68%
Neural Magic DeepSparse - C.C.R.5.I - A.M.S (items/sec)	34.3354	34.3542	34.4836
Normalized	99.57%	99.62%	100%
Neural Magic DeepSparse - C.C.R.5.I - S.S.S (items/sec)	33.6523	33.6395	33.6288
Normalized	100%	99.96%	99.93%
Neural Magic DeepSparse - N.T.C.D.m - A.M.S (items/sec)	18.6457	18.7697	18.8072
Normalized	99.14%	99.8%	100%
Neural Magic DeepSparse - N.T.C.D.m - S.S.S (items/sec)	17.1147	17.0586	17.0984
Normalized	100%	99.67%	99.9%
Neural Magic DeepSparse - C.S.9.P.Y.P - A.M.S (items/sec)	4.3781	4.3125	4.391
Normalized	99.71%	98.21%	100%
Neural Magic DeepSparse - C.S.9.P.Y.P - S.S.S (items/sec)	4.4416	4.4172	4.4042
Normalized	100%	99.45%	99.16%
Neural Magic DeepSparse - N.T.C.B.b.u.S - A.M.S (items/sec)	8.6291	8.6367	8.6032
Normalized	99.91%	100%	99.61%
Neural Magic DeepSparse - N.T.C.B.b.u.S - S.S.S (items/sec)	7.9995	8.0057	8.0153
Normalized	99.8%	99.88%	100%
Neural Magic DeepSparse - N.T.C.B.b.u.c - A.M.S (items/sec)	1.9802	1.9823	1.9885
Normalized	99.58%	99.69%	100%
Neural Magic DeepSparse - N.T.C.B.b.u.c - S.S.S (items/sec)	1.9825	1.9836	1.9828
Normalized	99.94%	100%	99.96%
KeyDB - GET - 50 (Reqs/sec)	965960	919540	908612
Normalized	100%	95.19%	94.06%
KeyDB - SET - 50 (Reqs/sec)	791390	800923	783883
Normalized	98.81%	100%	97.87%
KeyDB - GET - 100 (Reqs/sec)	909571	768805	856062
Normalized	100%	84.52%	94.12%
KeyDB - GET - 500 (Reqs/sec)	889079	742060	752604
Normalized	100%	83.46%	84.65%
KeyDB - GET - 900 (Reqs/sec)	865981	700074	711207
Normalized	100%	80.84%	82.13%

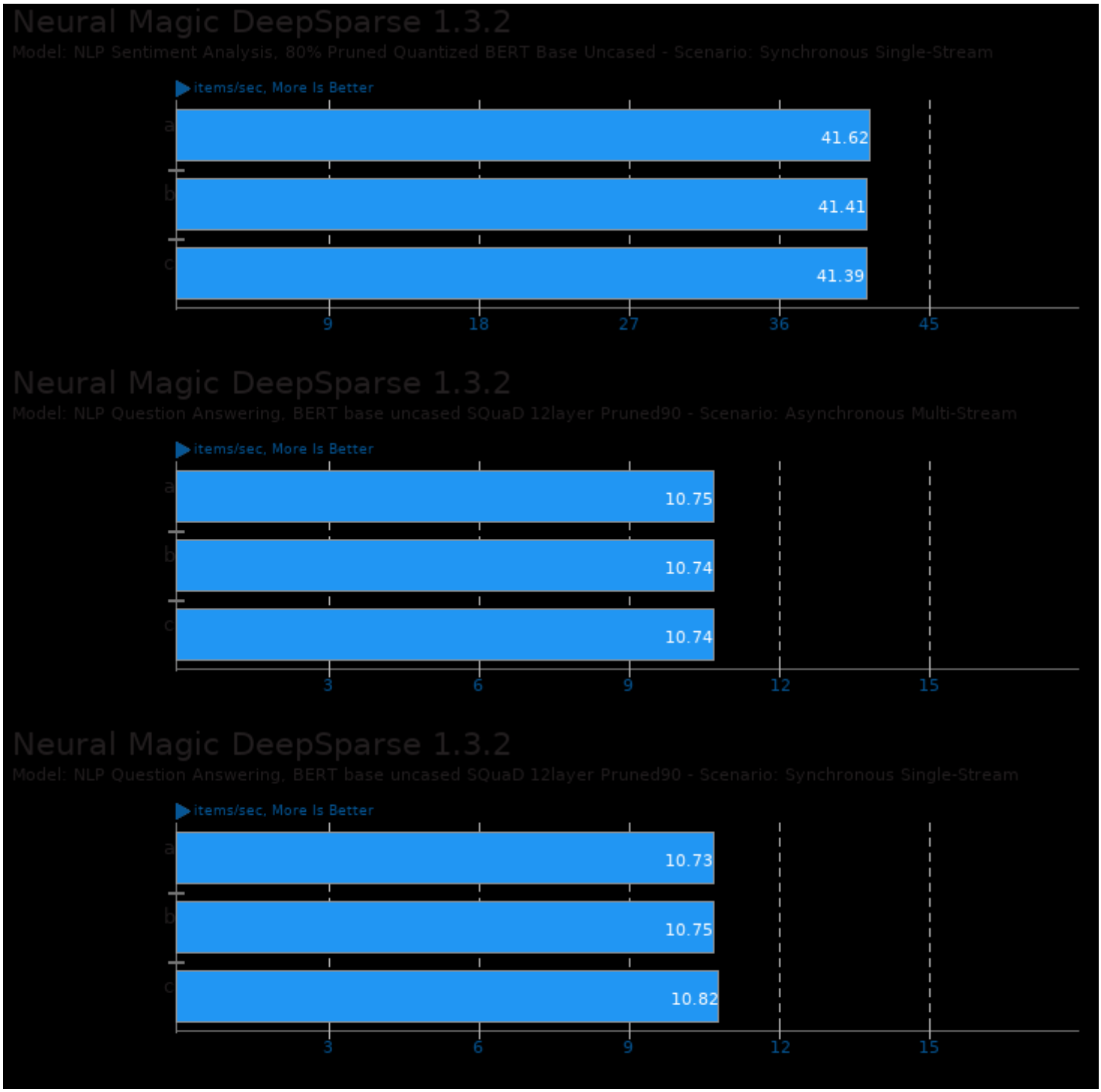
icelake deep

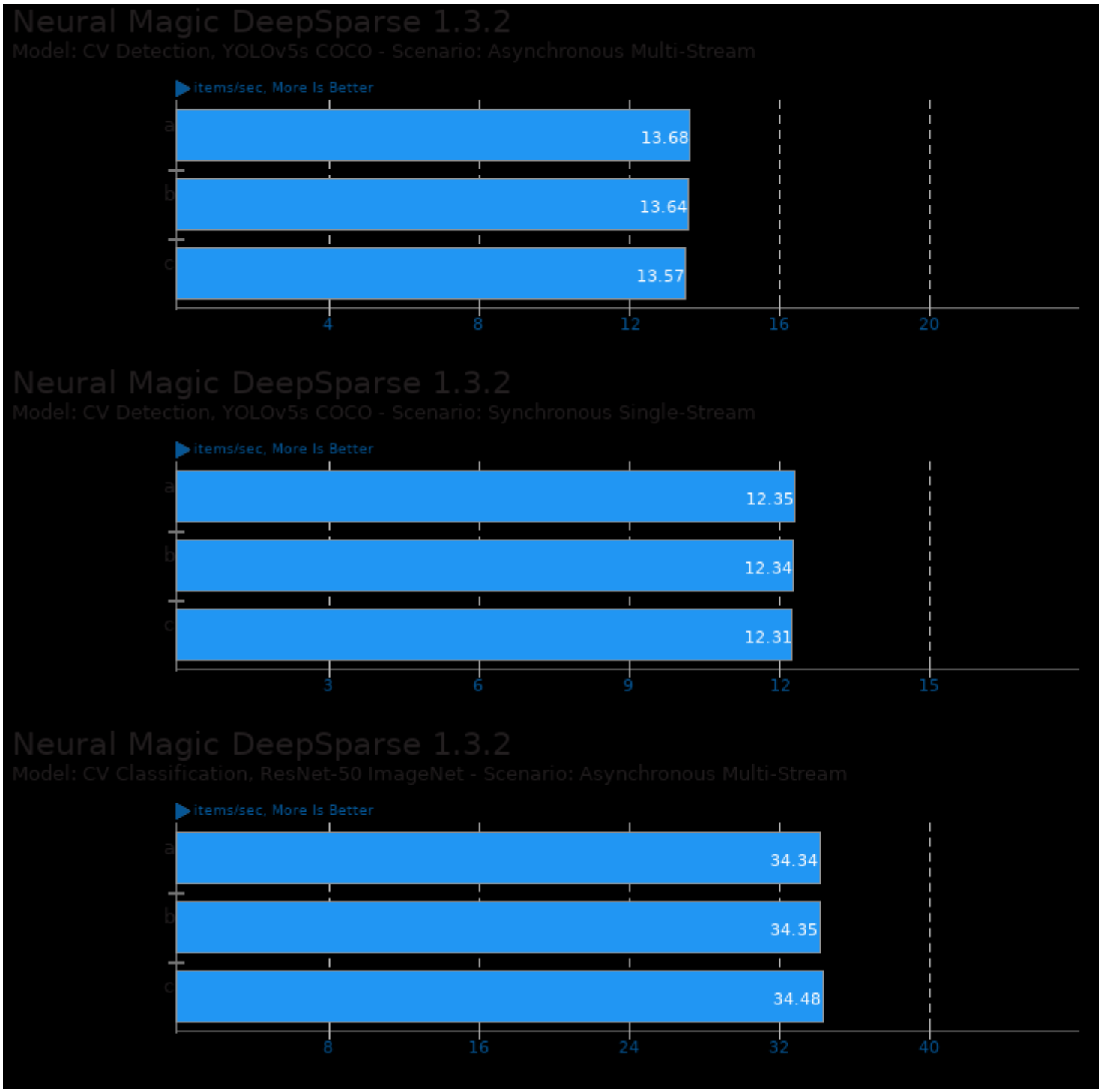
KeyDB - LPOP - 50 (Reqs/sec)	991179	649747	618965
Normalized	100%	65.55%	62.45%
KeyDB - SADD - 50 (Reqs/sec)	878272	768793	750593
Normalized	100%	87.53%	85.46%
KeyDB - SET - 100 (Reqs/sec)	815381	696563	677948
Normalized	100%	85.43%	83.14%
KeyDB - SET - 500 (Reqs/sec)	396372	641659	663183
Normalized	59.77%	96.75%	100%
KeyDB - SET - 900 (Reqs/sec)	703037	614002	609370
Normalized	100%	87.34%	86.68%
KeyDB - HMSET - 50 (Reqs/sec)	306581	333883	300564
Normalized	91.82%	100%	90.02%
KeyDB - LPOP - 100 (Reqs/sec)	910150	604354	616424
Normalized	100%	66.4%	67.73%
KeyDB - LPOP - 500 (Reqs/sec)	906109	604763	596154
Normalized	100%	66.74%	65.79%
KeyDB - LPOP - 900 (Reqs/sec)	809376	682436	642269
Normalized	100%	84.32%	79.35%
KeyDB - LPUSH - 50 (Reqs/sec)	643468	618728	644712
Normalized	99.81%	95.97%	100%
KeyDB - SADD - 100 (Reqs/sec)	814691	758576	759613
Normalized	100%	93.11%	93.24%
KeyDB - SADD - 500 (Reqs/sec)	739678	716723	735435
Normalized	100%	96.9%	99.43%
KeyDB - SADD - 900 (Reqs/sec)	743583	696806	679339
Normalized	100%	93.71%	91.36%
KeyDB - HMSET - 100 (Reqs/sec)	245915	252202	248416
Normalized	97.51%	100%	98.5%
KeyDB - HMSET - 500 (Reqs/sec)	241320	238657	235957
Normalized	100%	98.9%	97.78%
KeyDB - HMSET - 900 (Reqs/sec)	235377	234181	233904
Normalized	100%	99.49%	99.37%
KeyDB - LPUSH - 100 (Reqs/sec)	605936	596282	603624
Normalized	100%	98.41%	99.62%
KeyDB - LPUSH - 500 (Reqs/sec)	558753	553452	560846
Normalized	99.63%	98.68%	100%
KeyDB - LPUSH - 900 (Reqs/sec)	547585	535200	537872
Normalized	100%	97.74%	98.23%
Neural Magic DeepSparse - N.D.C.o.b.u.o.l - A.M.S (ms/batch)	1008	1006	1001
Normalized	99.38%	99.54%	100%
Neural Magic DeepSparse - N.D.C.o.b.u.o.l - S.S.S (ms/batch)	502.7354	503.5758	503.499
Normalized	100%	99.83%	99.85%
Neural Magic DeepSparse - N.S.A.8.P.Q.B.B.U - A.M.S (ms/batch)	45.7066	47.0387	45.9597
Normalized	100%	97.17%	99.45%
Neural Magic DeepSparse - N.S.A.8.P.Q.B.B.U - S.S.S (ms/batch)	24.0178	24.1386	24.1502
Normalized	100%	99.5%	99.45%

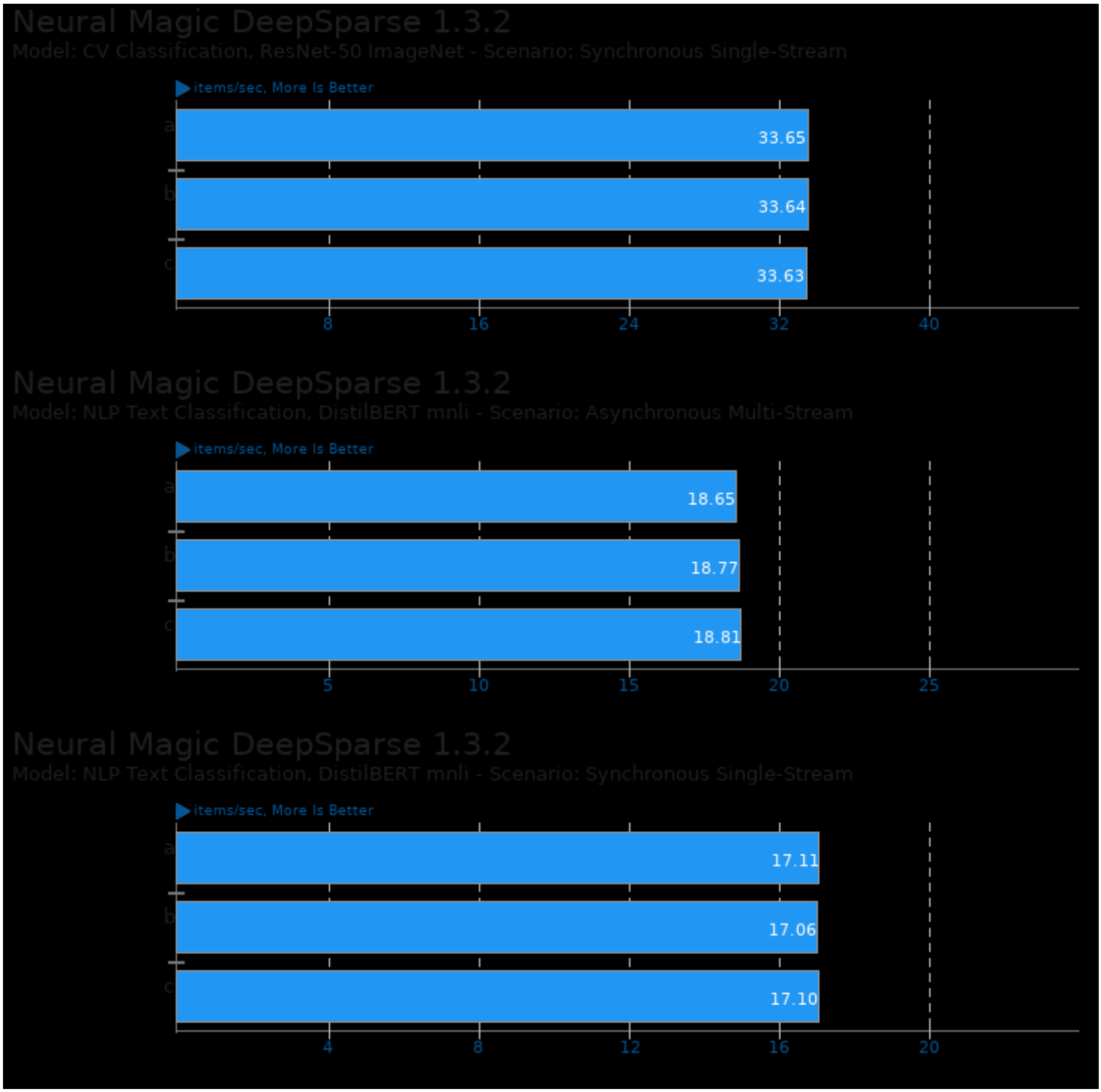
icelake deep

Neural Magic DeepSparse - N.Q.A.B.b.u.S.1.P - A.M.S (ms/batch)	185.64	186.0784	186.122
Normalized	100%	99.76%	99.74%
Neural Magic DeepSparse - N.Q.A.B.b.u.S.1.P - S.S.S (ms/batch)	93.1713	93.0114	92.3732
Normalized	99.14%	99.31%	100%
Neural Magic DeepSparse - C.D.Y.C - A.M.S (ms/batch)	146.1429	146.6365	147.4004
Normalized	100%	99.66%	99.15%
Neural Magic DeepSparse - C.D.Y.C - S.S.S (ms/batch)	80.9337	81.0418	81.1917
Normalized	100%	99.87%	99.68%
Neural Magic DeepSparse - C.C.R.5.I - A.M.S (ms/batch)	58.2053	58.1527	57.9441
Normalized	99.55%	99.64%	100%
Neural Magic DeepSparse - C.C.R.5.I - S.S.S (ms/batch)	29.7036	29.7149	29.7237
Normalized	100%	99.96%	99.93%
Neural Magic DeepSparse - N.T.C.D.m - A.M.S (ms/batch)	107.1765	106.4621	106.2527
Normalized	99.14%	99.8%	100%
Neural Magic DeepSparse - N.T.C.D.m - S.S.S (ms/batch)	58.4197	58.6125	58.4757
Normalized	100%	99.67%	99.9%
Neural Magic DeepSparse - C.S.9.P.Y.P - A.M.S (ms/batch)	456.7458	463.5304	455.3891
Normalized	99.7%	98.24%	100%
Neural Magic DeepSparse - C.S.9.P.Y.P - S.S.S (ms/batch)	225.1197	226.366	227.0317
Normalized	100%	99.45%	99.16%
Neural Magic DeepSparse - N.T.C.B.b.u.S - A.M.S (ms/batch)	231.3358	231.2106	232.0997
Normalized	99.95%	100%	99.62%
Neural Magic DeepSparse - N.T.C.B.b.u.S - S.S.S (ms/batch)	124.9982	124.9007	124.7501
Normalized	99.8%	99.88%	100%
Neural Magic DeepSparse - N.T.C.B.b.u.c - A.M.S (ms/batch)	1008	1008	1005
Normalized	99.71%	99.73%	100%
Neural Magic DeepSparse - N.T.C.B.b.u.c - S.S.S (ms/batch)	504.4064	504.1199	504.3356
Normalized	99.94%	100%	99.96%

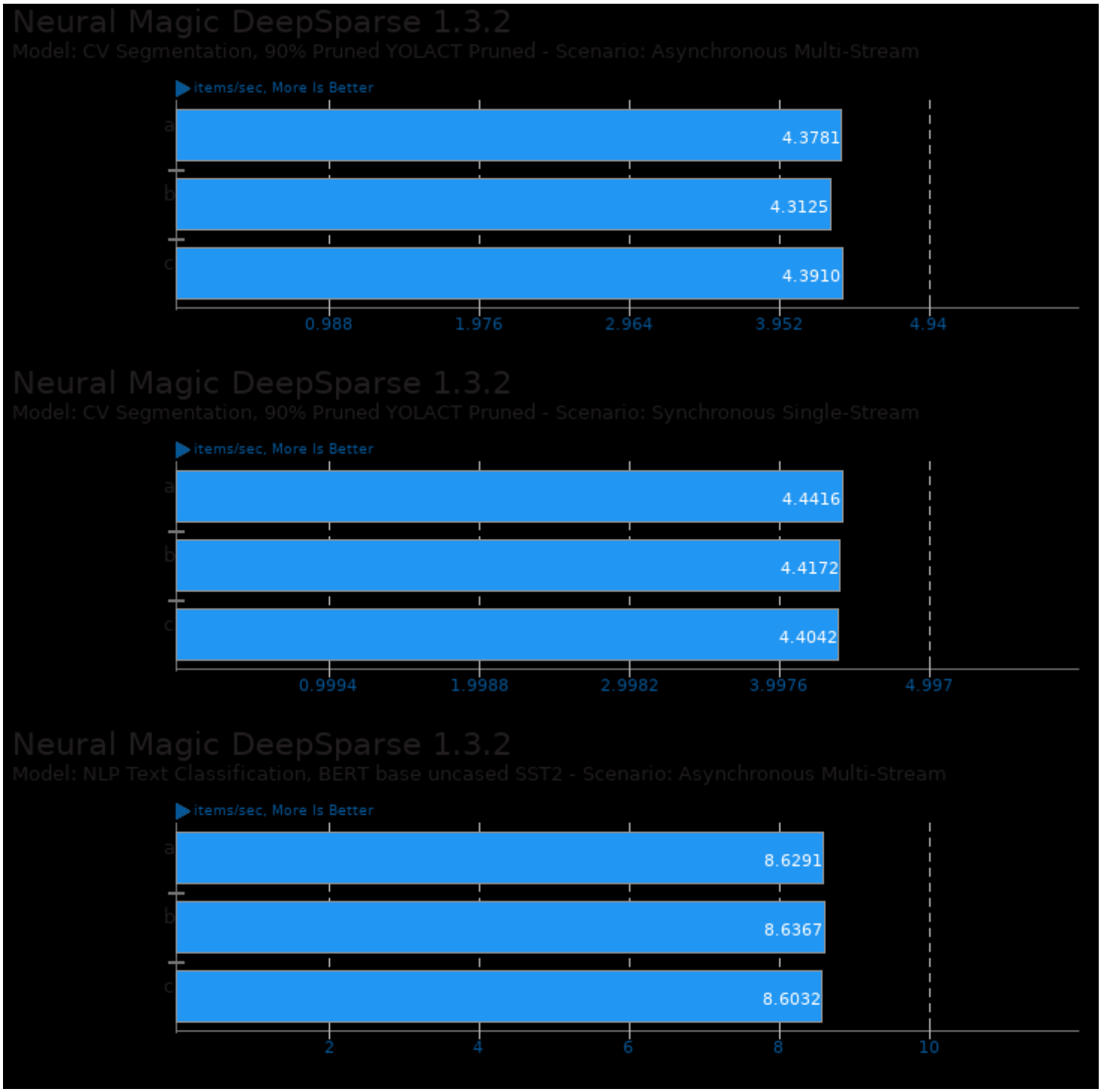

Neural Magic DeepSparse 1.3.2
Model: NLP Document Classification, oBERT base uncased on IMDB - Scenario: Asynchronous Multi-Stream
items/sec, More Is Better
a
b
c
1.9804
1.9855
1.9956
0.449
0.898
1.347
1.796
2.245
Neural Magic DeepSparse 1.3.2
Model: NLP Document Classification, oBERT base uncased on IMDB - Scenario: Synchronous Single-Stream
items/sec, More Is Better
a
b
c
1.9891
1.9858
1.9861
0.4475
0.895
1.3425
1.79
2.2375
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
43.72
42.47
43.49
10
20
30
40
50

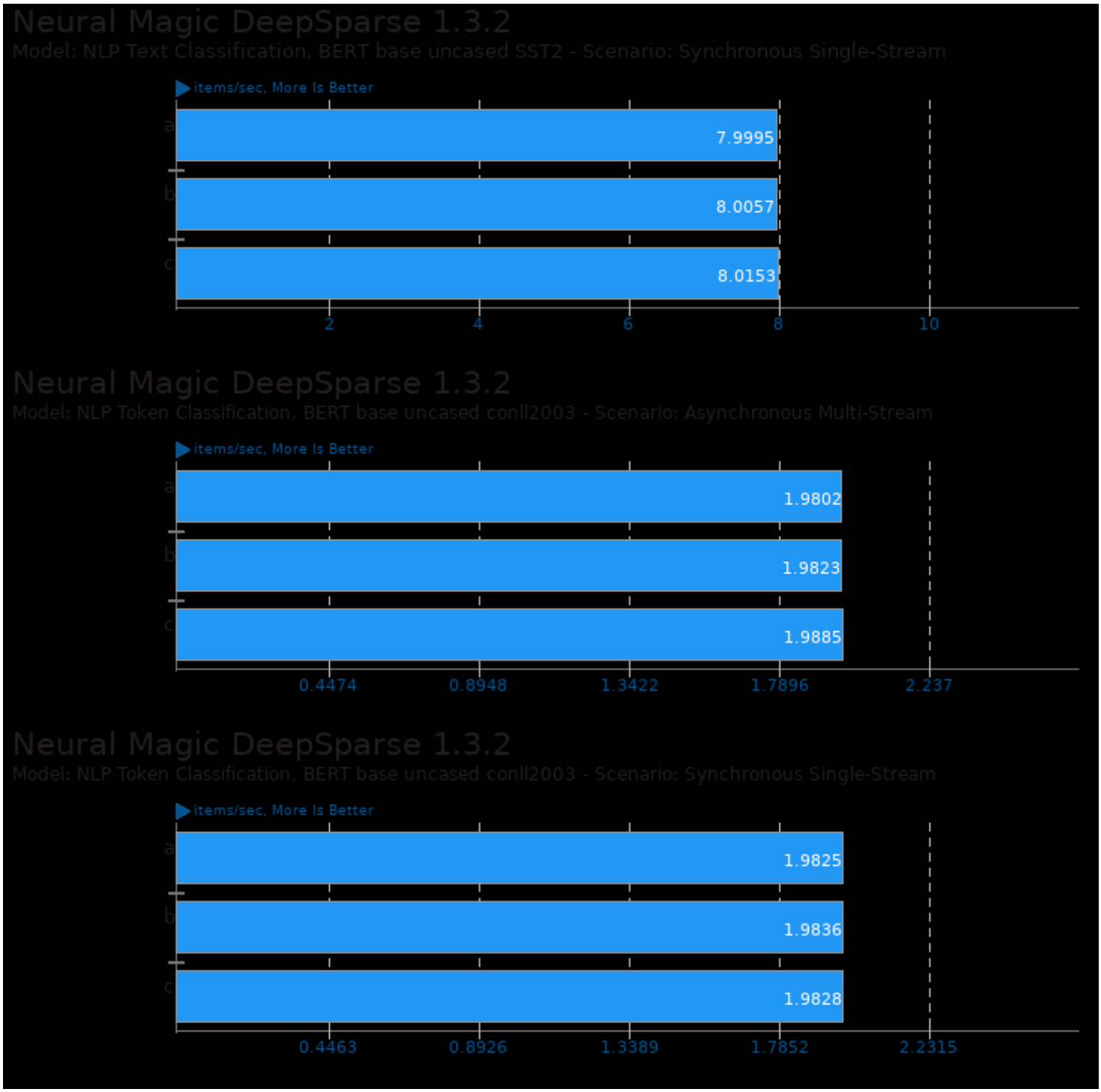

Neural Magic DeepSparse 1.3.2
Model: NLP Sentiment Analysis, 80% Pruned Quantized BERT Base Uncased - Scenario: Synchronous Single-Stream
items/sec, More Is Better
a
b
c
41.62
41.41
41.39
9
18
27
36
45
Neural Magic DeepSparse 1.3.2
Model: NLP Question Answering, BERT base uncased SQuaD 12layer Pruned90 - Scenario: Asynchronous Multi-Stream
items/sec, More Is Better
a
b
c
10.75
10.74
10.74
3
6
9
12
15
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
10.73
10.75
10.82
3
6
9
12
15

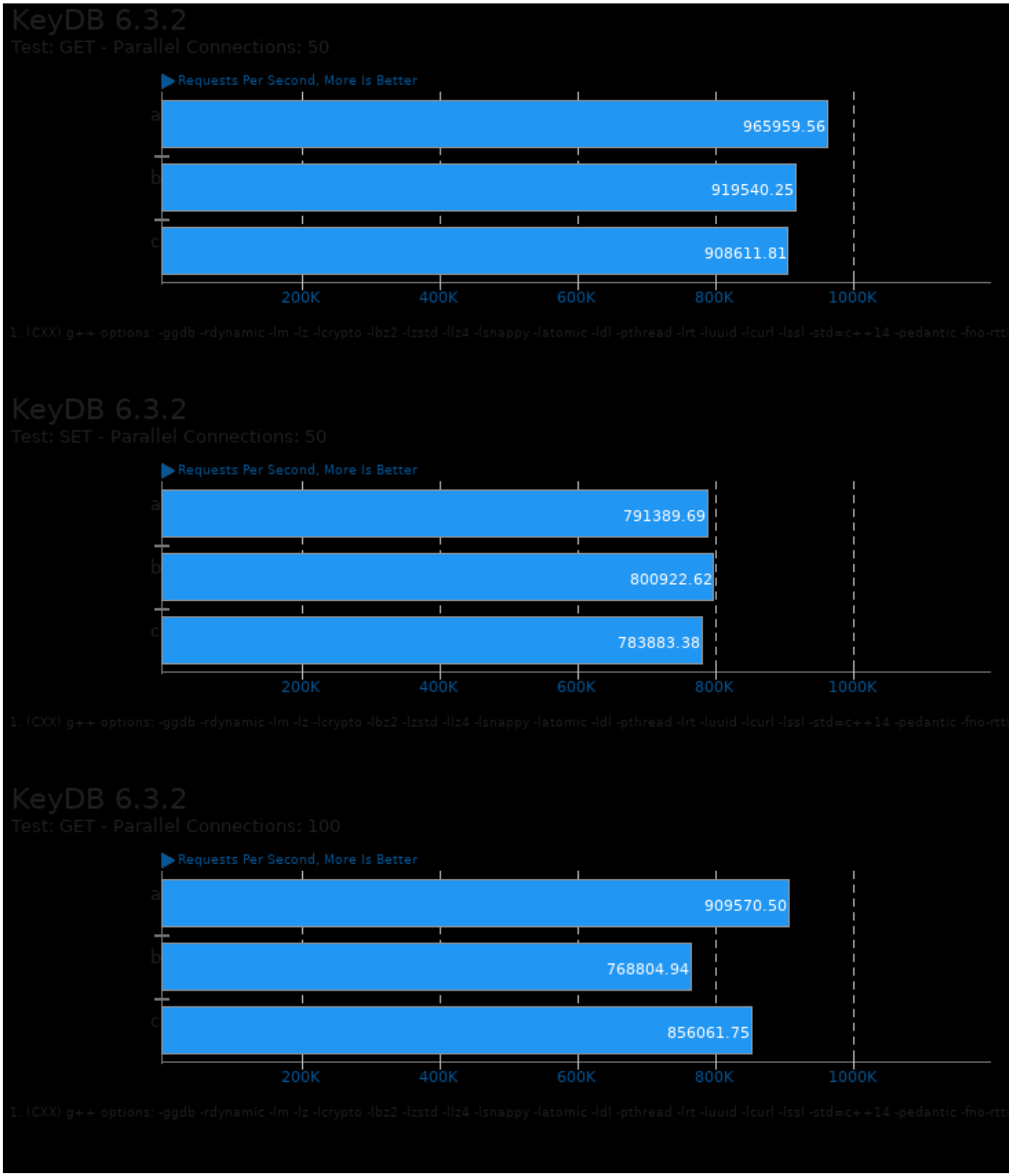

Neural Magic DeepSparse 1.3.2
Model: CV Detection, YOLOv5s COCO - Scenario: Asynchronous Multi-Stream
items/sec, More Is Better
a
b
c
13.68
13.64
13.57
4
8
12
16
20
Neural Magic DeepSparse 1.3.2
Model: CV Detection, YOLOv5s COCO - Scenario: Synchronous Single-Stream
items/sec, More Is Better
a
b
c
12.35
12.34
12.31
3
6
9
12
15
Neural Magic DeepSparse 1.3.2
Model: CV Classification, ResNet-50 ImageNet - Scenario: Asynchronous Multi-Stream
items/sec, More Is Better
a
b
c
34.34
34.35
34.48
8
16
24
32
40

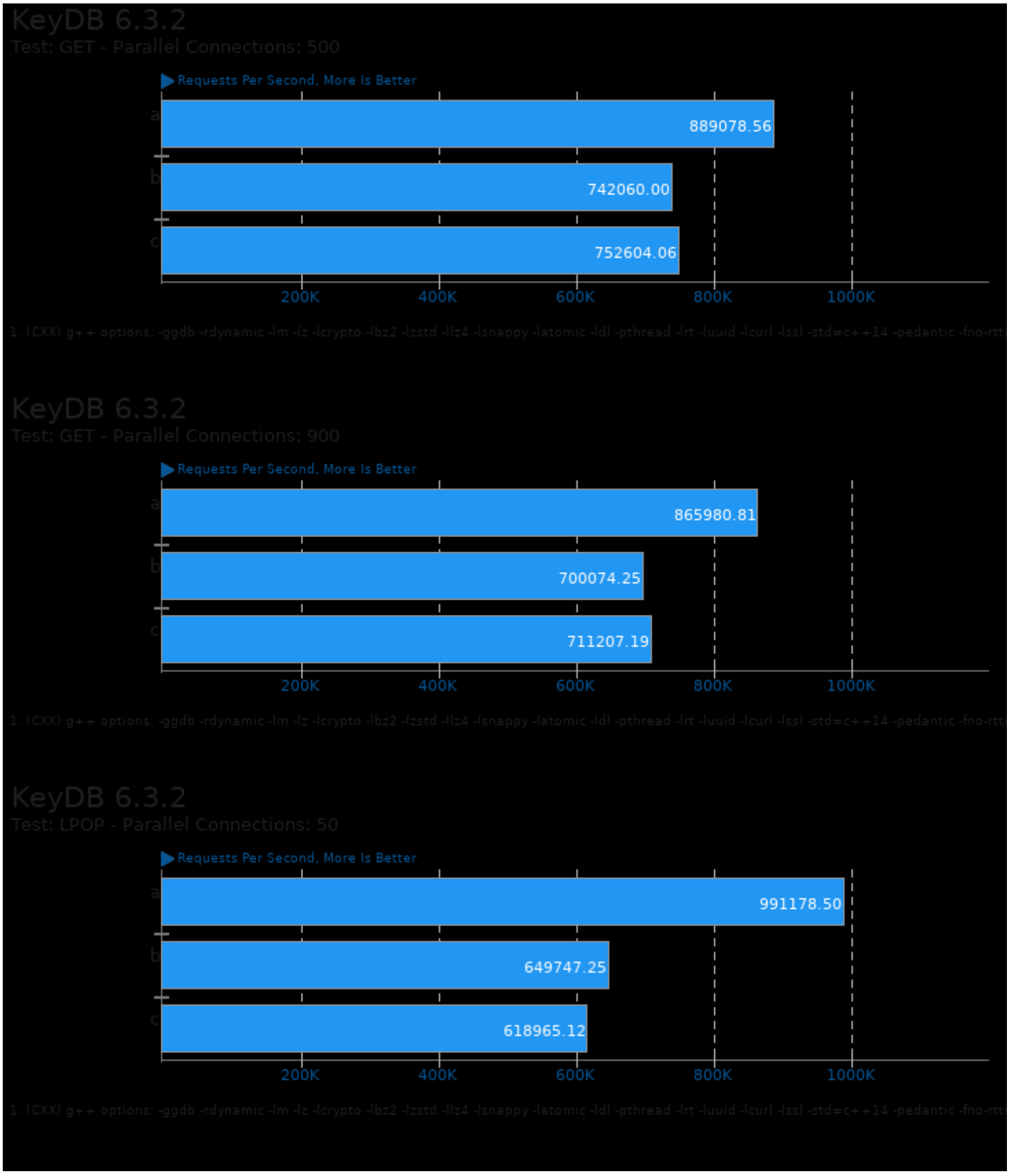

Neural Magic DeepSparse 1.3.2
Model: CV Classification, ResNet-50 ImageNet - Scenario: Synchronous Single-Stream
items/sec, More Is Better
a
b
c
33.65
33.64
33.63
8
16
24
32
40
Neural Magic DeepSparse 1.3.2
Model: NLP Text Classification, DistilBERT mnli - Scenario: Asynchronous Multi-Stream
items/sec, More Is Better
a
b
c
18.65
18.77
18.81
5
10
15
20
25
Neural Magic DeepSparse 1.3.2
Model: NLP Text Classification, DistilBERT mnli - Scenario: Synchronous Single-Stream
items/sec, More Is Better
a
b
c
17.11
17.06
17.10
4
8
12
16
20

icelake deep



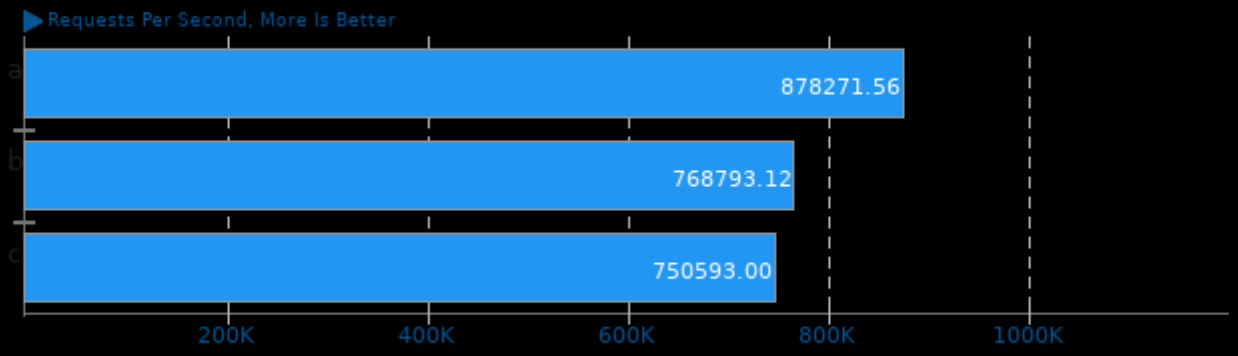

Neural Magic DeepSparse 1.3.2
Model: NLP Text Classification, BERT base uncased SST2 - Scenario: Synchronous Single-Stream
items/sec, More Is Better
a
b
c
7.9995
8.0057
8.0153
2
4
6
8
10
Neural Magic DeepSparse 1.3.2
Model: NLP Token Classification, BERT base uncased conll2003 - Scenario: Asynchronous Multi-Stream
items/sec, More Is Better
a
b
c
1.9802
1.9823
1.9885
0.4474
0.8948
1.3422
1.7896
2.237
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
1.9825
1.9836
1.9828
0.4463
0.8926
1.3389
1.7852
2.2315


KeyDB 6.3.2
Test: GET - Parallel Connections: 50
Requests Per Second, More Is Better
a
b
c
965959.56
919540.25
908611.81
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: 50
Requests Per Second, More Is Better
a
b
c
791389.69
800922.62
783883.38
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
909570.50
768804.94
856061.75
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: 500
Requests Per Second, More Is Better
a
889078.56
b
742060.00
c
752604.06
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
865980.81
b
700074.25
c
711207.19
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
991178.50
b
649747.25
c
618965.12
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

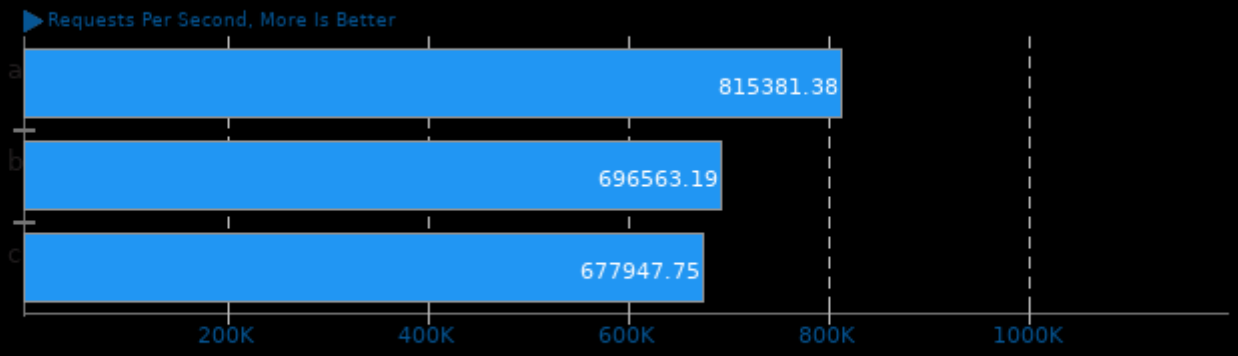
Test: SADD - Parallel Connections: 50



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-rtti

KeyDB 6.3.2

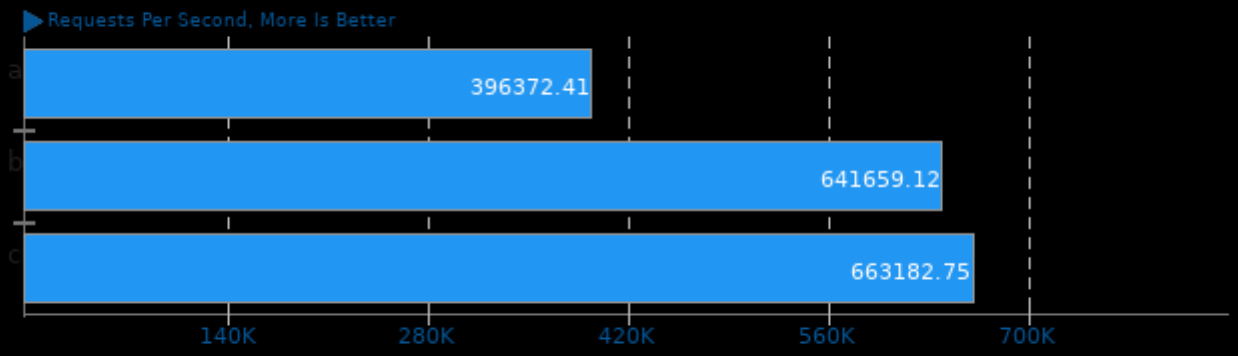
Test: SET - Parallel Connections: 100



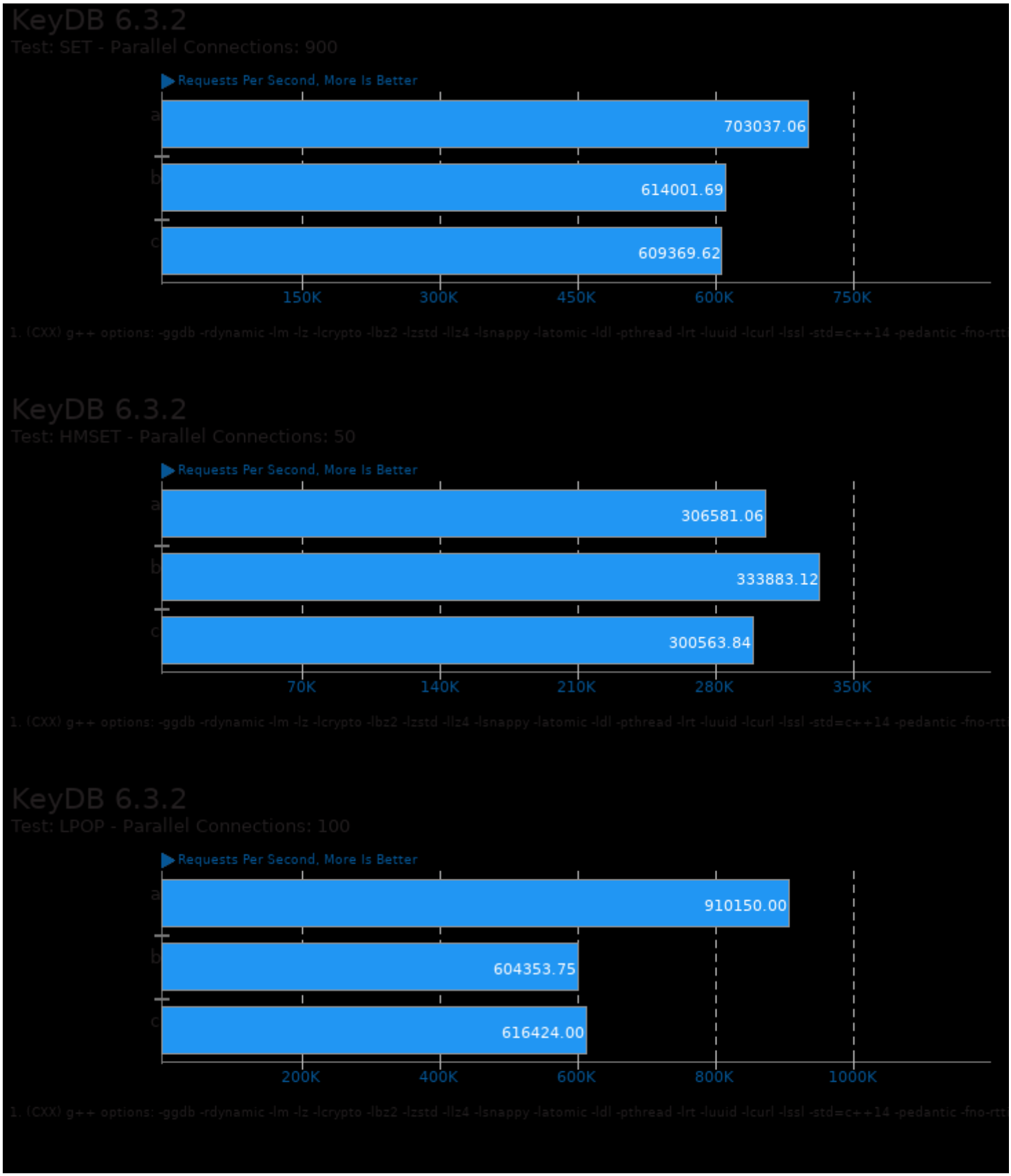
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtti

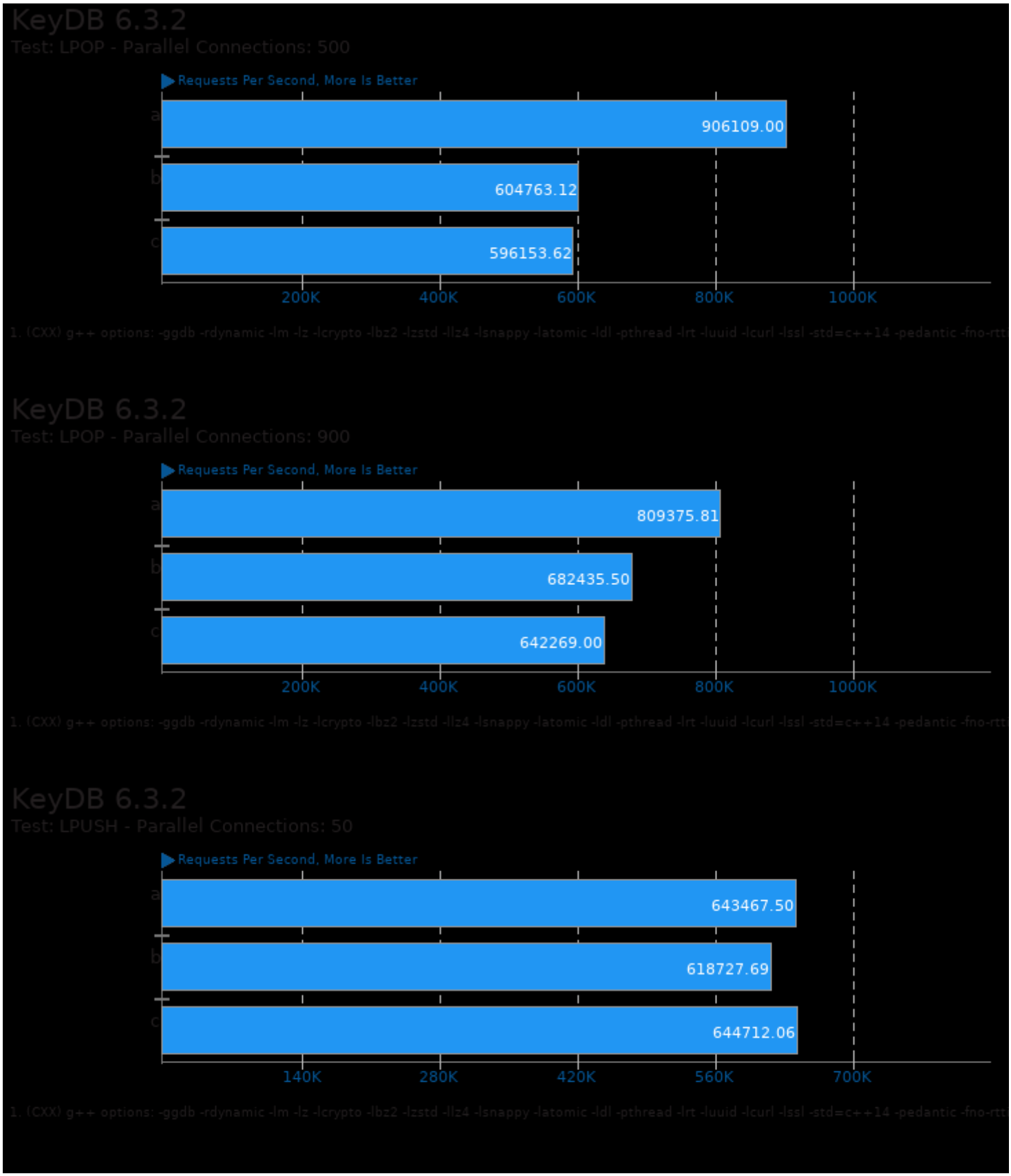
KeyDB 6.3.2

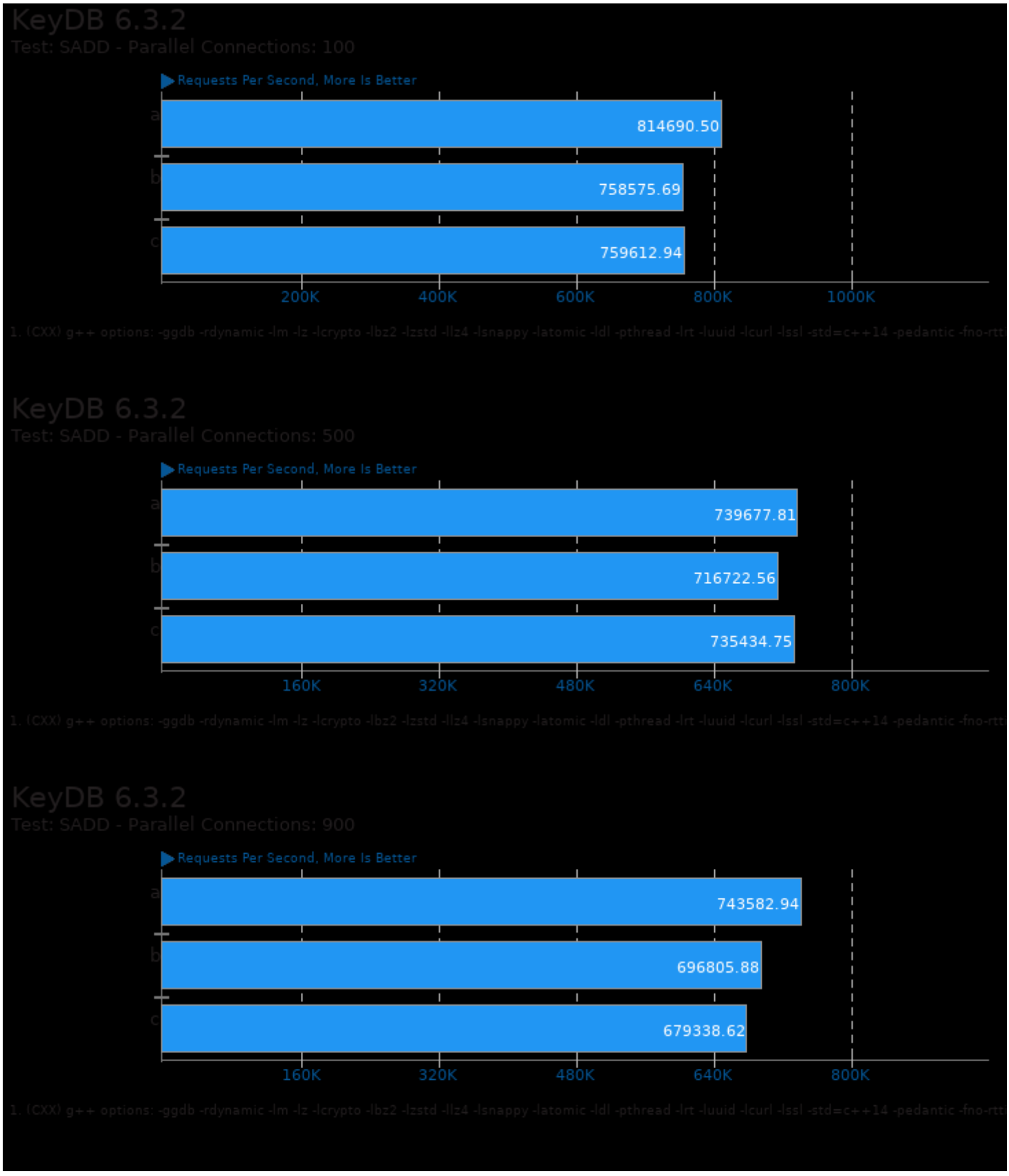
Test: SET - Parallel Connections: 500

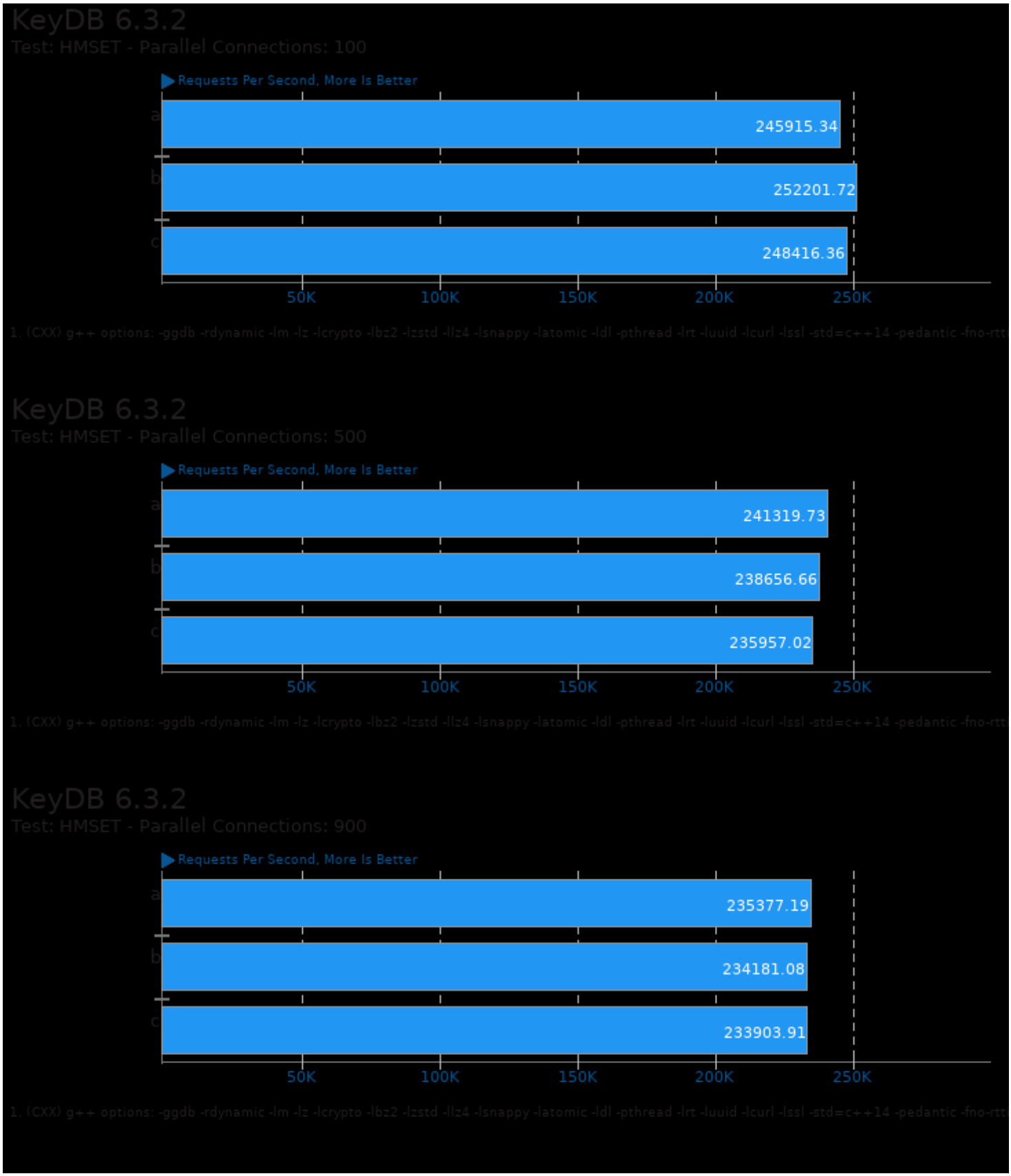


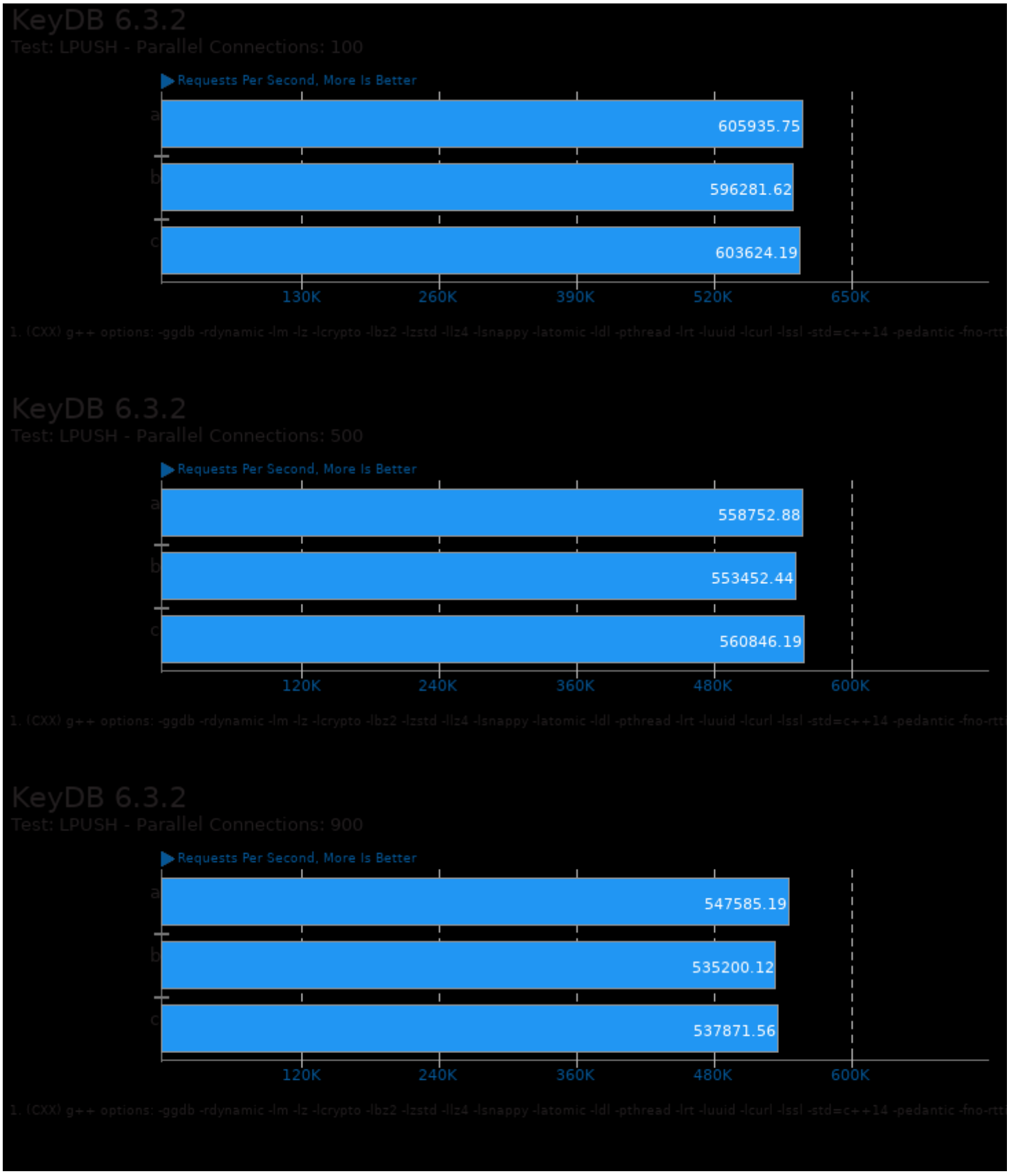
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtti

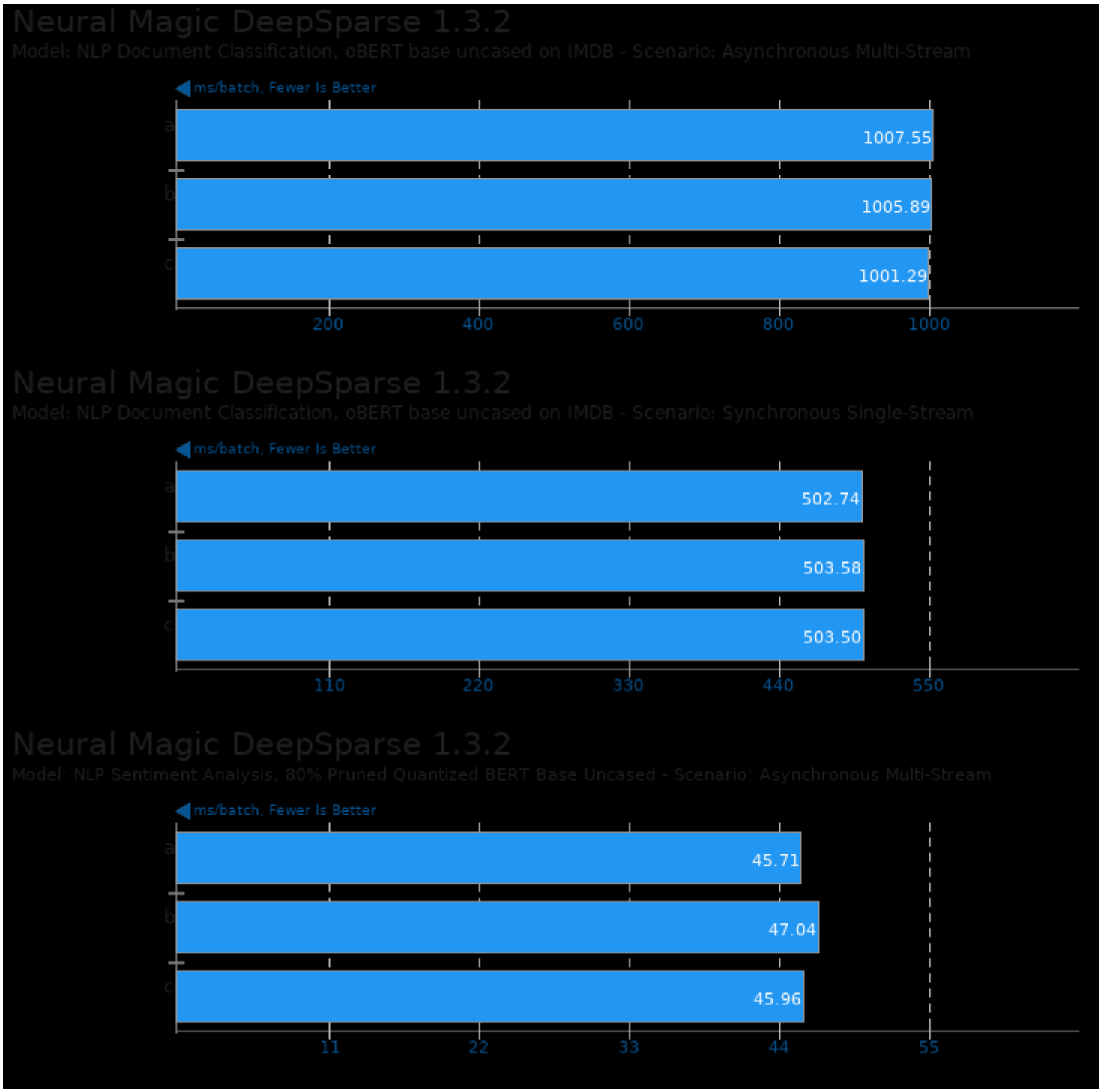

KeyDB 6.3.2
Test: SET - Parallel Connections: 900
Requests Per Second, More Is Better
a
b
c
703037.06
614001.69
609369.62
150K
300K
450K
600K
750K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtt
KeyDB 6.3.2
Test: HMSET - Parallel Connections: 50
Requests Per Second, More Is Better
a
b
c
306581.06
333883.12
300563.84
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: LPOP - Parallel Connections: 100
Requests Per Second, More Is Better
a
b
c
910150.00
604353.75
616424.00
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

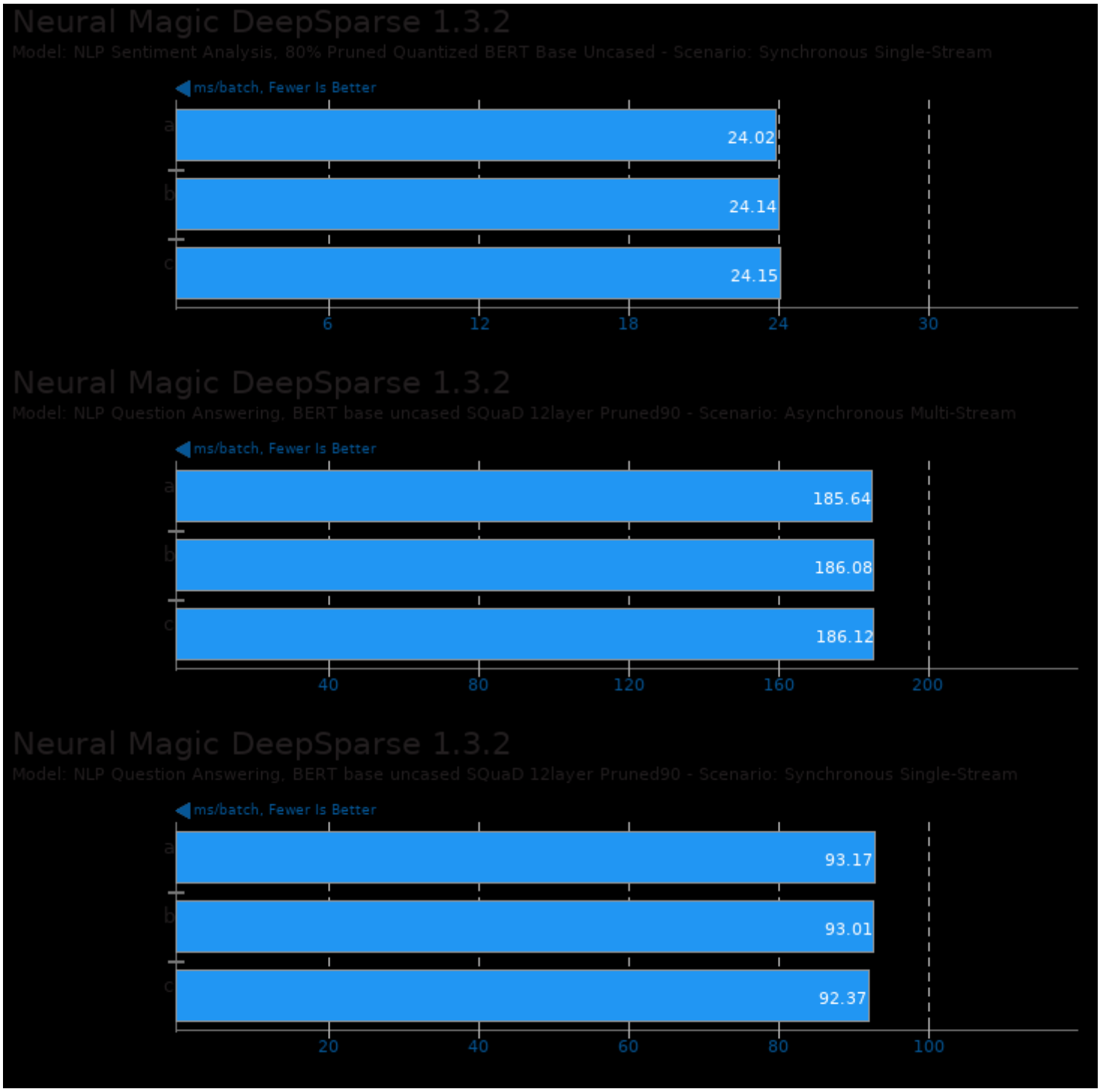

KeyDB 6.3.2
Test: LPOP - Parallel Connections: 500
Requests Per Second, More Is Better
a
b
c
906109.00
604763.12
596153.62
200K
400K
600K
800K
1000K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtt
KeyDB 6.3.2
Test: LPOP - Parallel Connections: 900
Requests Per Second, More Is Better
a
b
c
809375.81
682435.50
642269.00
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: LPUSH - Parallel Connections: 50
Requests Per Second, More Is Better
a
b
c
643467.50
618727.69
644712.06
140K
280K
420K
560K
700K
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

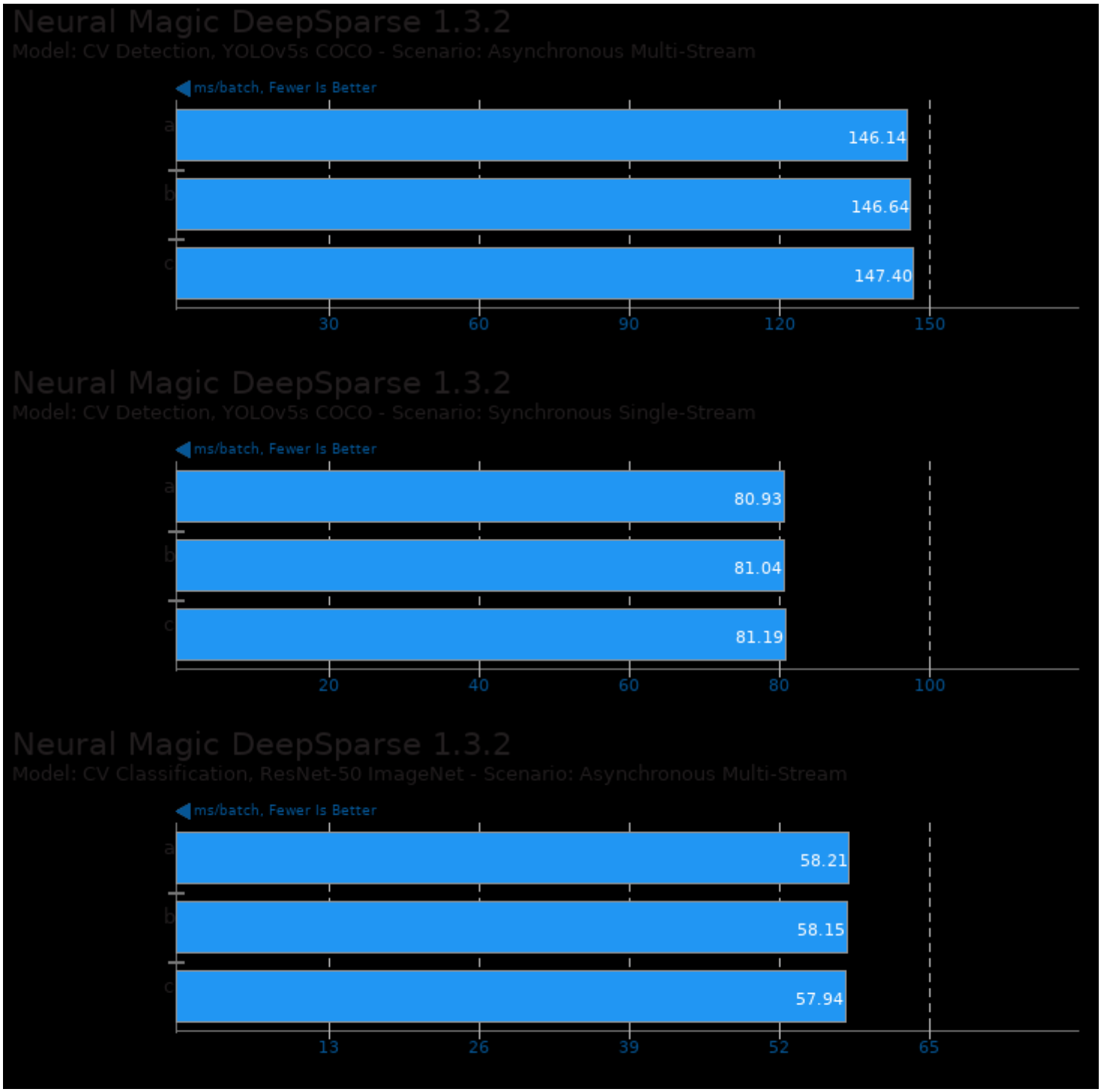

KeyDB 6.3.2
Test: SADD - Parallel Connections: 100
Requests Per Second, More Is Better
a
814690.50
b
758575.69
c
759612.94
200K
400K
600K
800K
1000K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtt
KeyDB 6.3.2
Test: SADD - Parallel Connections: 500
Requests Per Second, More Is Better
a
739677.81
b
716722.56
c
735434.75
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
KeyDB 6.3.2
Test: SADD - Parallel Connections: 900
Requests Per Second, More Is Better
a
743582.94
b
696805.88
c
679338.62
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

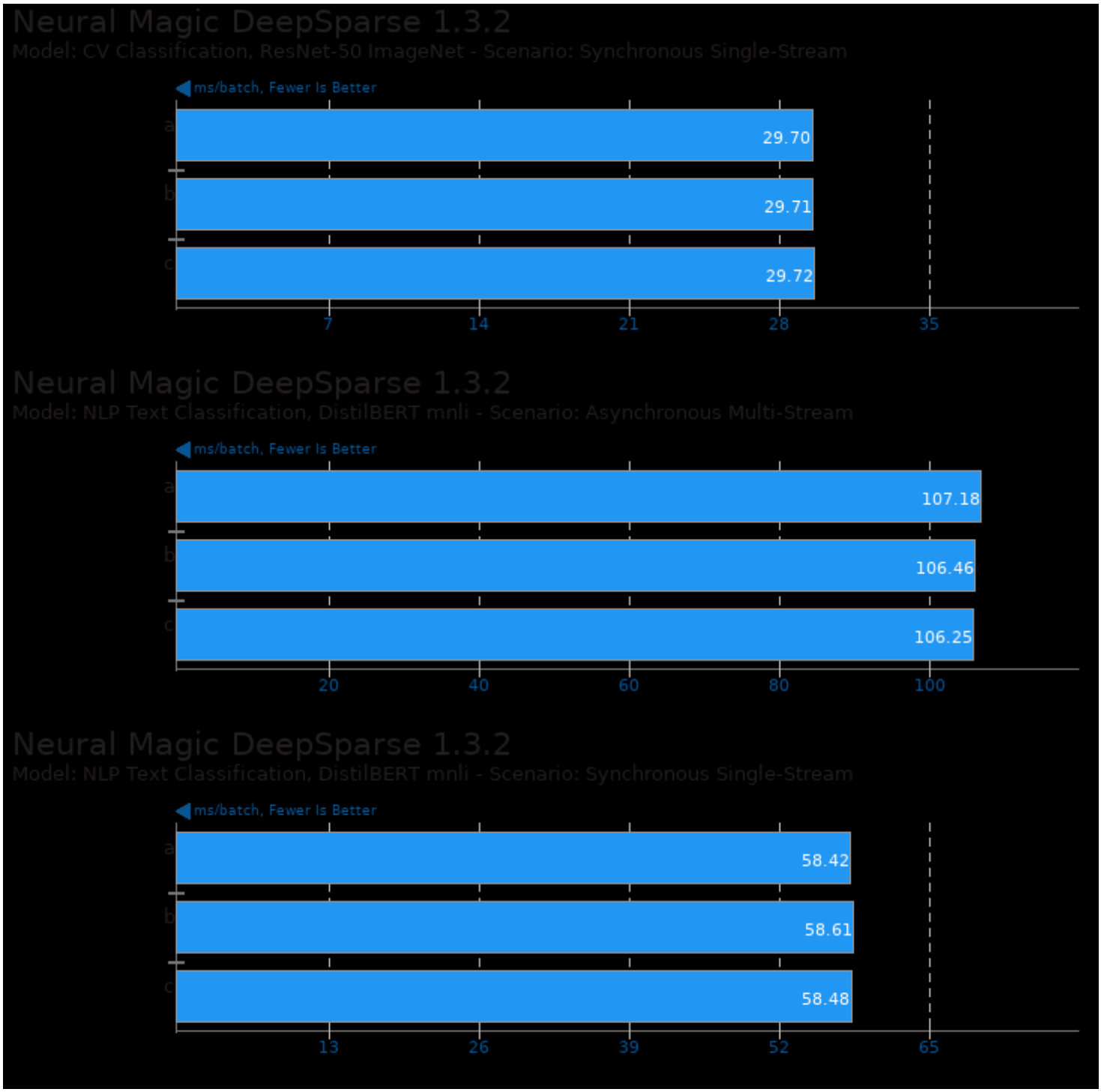

KeyDB 6.3.2
Test: HMSET - Parallel Connections: 100
Requests Per Second, More Is Better
a
b
c
245915.34
252201.72
248416.36
50K
100K
150K
200K
250K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtt
KeyDB 6.3.2
Test: HMSET - Parallel Connections: 500
Requests Per Second, More Is Better
a
b
c
241319.73
238656.66
235957.02
50K
100K
150K
200K
250K
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
235377.19
234181.08
233903.91
50K
100K
150K
200K
250K
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

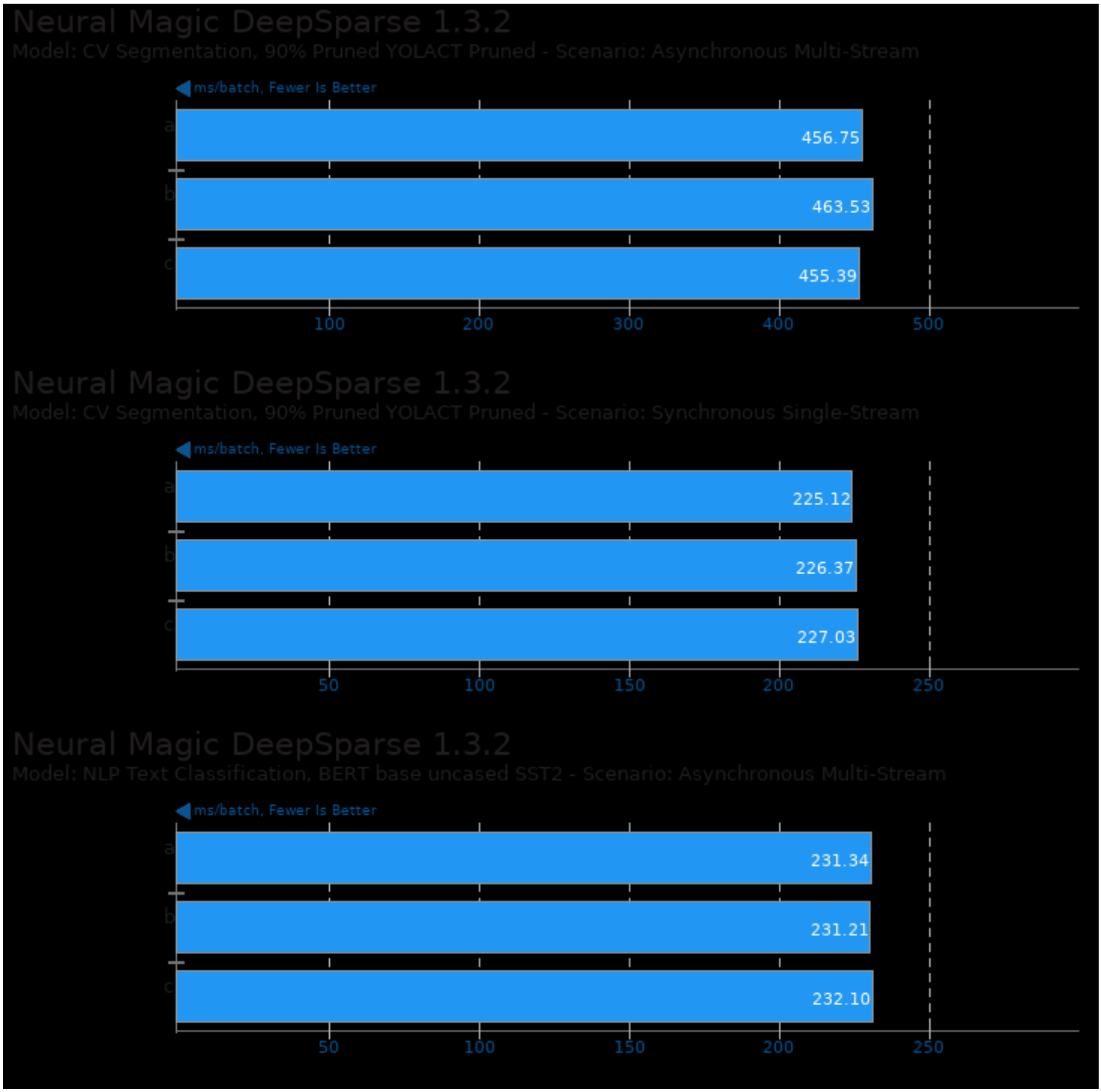

KeyDB 6.3.2
Test: LPUSH - Parallel Connections: 100
Requests Per Second, More Is Better
a
605935.75
b
596281.62
c
603624.19
130K
260K
390K
520K
650K
1. (CXX) g++ options: -ggdb -rdynamic -lm -lz -lcrypto -lbz2 -lzstd -llz4 -lsnappy -latomic -ldl -pthread -lrt -luuid -lcurl -lssl -std=c++14 -pedantic -fno-rtt
KeyDB 6.3.2
Test: LPUSH - Parallel Connections: 500
Requests Per Second, More Is Better
a
558752.88
b
553452.44
c
560846.19
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: LPUSH - Parallel Connections: 900
Requests Per Second, More Is Better
a
547585.19
b
535200.12
c
537871.56
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

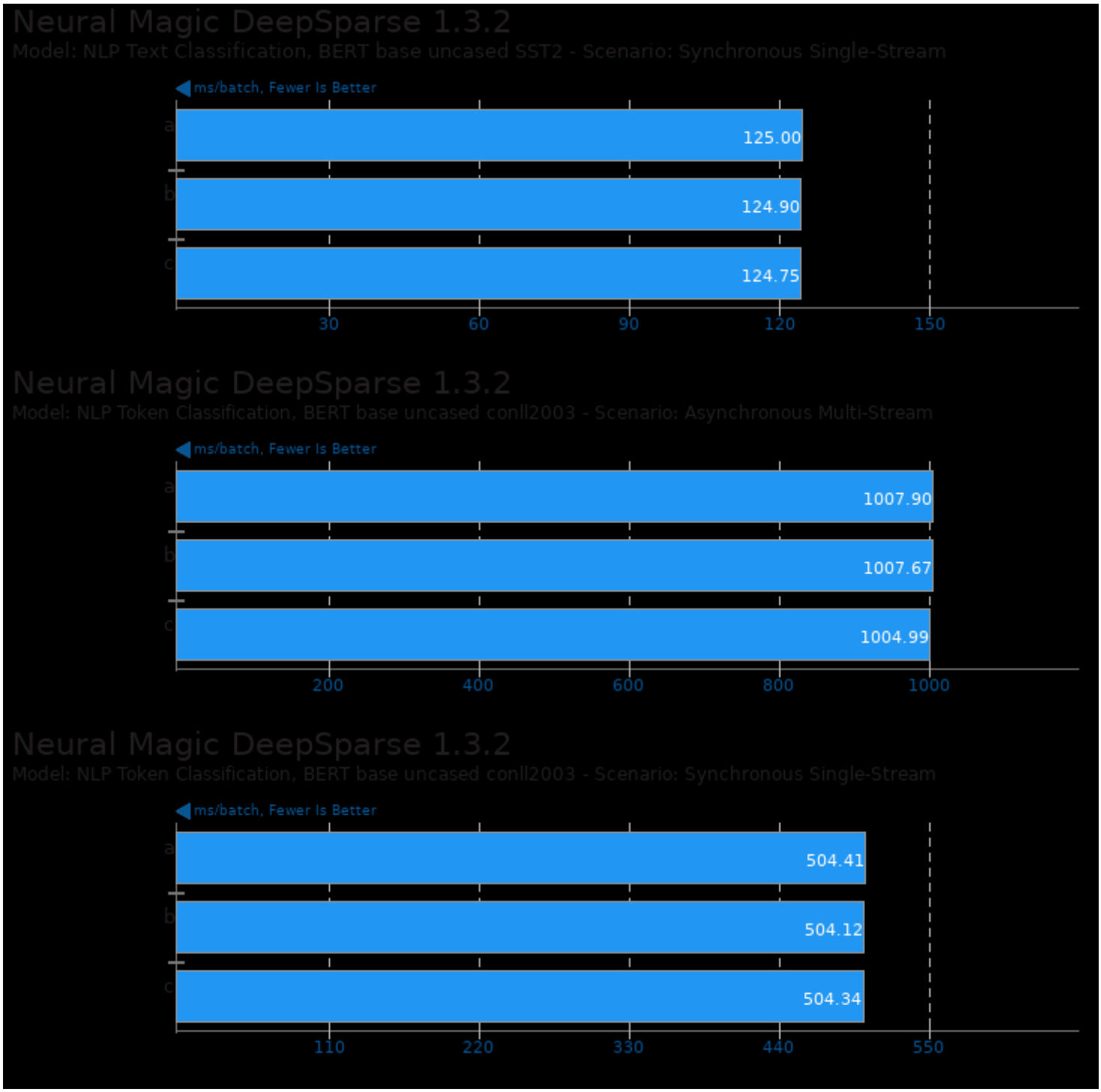

Neural Magic DeepSparse 1.3.2
Model: NLP Document Classification, oBERT base uncased on IMDB - Scenario: Asynchronous Multi-Stream
ms/batch, Fewer Is Better
a
b
c
1007.55
1005.89
1001.29
200
400
600
800
1000
Neural Magic DeepSparse 1.3.2
Model: NLP Document Classification, oBERT base uncased on IMDB - Scenario: Synchronous Single-Stream
ms/batch, Fewer Is Better
a
b
c
502.74
503.58
503.50
110
220
330
440
550
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
45.71
47.04
45.96
11
22
33
44
55


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
24.02
24.14
24.15
6
12
18
24
30
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
185.64
186.08
186.12
40
80
120
160
200
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
93.17
93.01
92.37
20
40
60
80
100


Neural Magic DeepSparse 1.3.2
Model: CV Detection, YOLOv5s COCO - Scenario: Asynchronous Multi-Stream
ms/batch, Fewer Is Better
a
b
c
146.14
146.64
147.40
30
60
90
120
150
Neural Magic DeepSparse 1.3.2
Model: CV Detection, YOLOv5s COCO - Scenario: Synchronous Single-Stream
ms/batch, Fewer Is Better
a
b
c
80.93
81.04
81.19
20
40
60
80
100
Neural Magic DeepSparse 1.3.2
Model: CV Classification, ResNet-50 ImageNet - Scenario: Asynchronous Multi-Stream
ms/batch, Fewer Is Better
a
b
c
58.21
58.15
57.94
13
26
39
52
65


Neural Magic DeepSparse 1.3.2
Model: CV Classification, ResNet-50 ImageNet - Scenario: Synchronous Single-Stream
ms/batch, Fewer Is Better
a
b
c
29.70
29.71
29.72
7
14
21
28
35
Neural Magic DeepSparse 1.3.2
Model: NLP Text Classification, DistilBERT mnli - Scenario: Asynchronous Multi-Stream
ms/batch, Fewer Is Better
a
b
c
107.18
106.46
106.25
20
40
60
80
100
Neural Magic DeepSparse 1.3.2
Model: NLP Text Classification, DistilBERT mnli - Scenario: Synchronous Single-Stream
ms/batch, Fewer Is Better
a
b
c
58.42
58.61
58.48
13
26
39
52
65


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
456.75
463.53
455.39
100
200
300
400
500
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
225.12
226.37
227.03
50
100
150
200
250
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
231.34
231.21
232.10
50
100
150
200
250



This file was automatically generated via the Phoronix Test Suite benchmarking software on Friday, 29 March 2024 01:47.