EPJ Web of Conferences **295**, 01044 (2024) https://doi.org/10.1051/epjconf/202429501044
CHEP 2023

Predicting Resource Utilization Trends with Southern California Petabyte Scale Cache

Caitlin Sim[1,*], *Kesheng* Wu[2,**], *Alex* Sim[2,***], *Inder* Monga[3,****], *Chin* Guok[3,†], *Damian* Hazen[3,‡], *Frank* Würthwein[4,§], *Diego* Davila[4,¶], *Harvey* Newman[5,‖], and *Justas* Balcas[5,**]

[1]University of California at Berkeley, Berkeley, CA, USA
[2]Lawrence Berkeley National Laboratory, Berkeley, CA, USA
[3]Energy Sciences Network, Berkeley, CA, USA
[4]University of California at San Diego, La Jolla, CA, USA
[5]California Institute of Technology, Pasadena, CA, USA

Abstract. Large community of high-energy physicists share their data all around world making it necessary to ship a large number of files over wide-area networks. Regional disk caches such as the Southern California Petabyte Scale Cache have been deployed to reduce the data access latency. We observe that about 94% of the requested data volume were served from this cache, without remote transfers, between Sep. 2022 and July 2023. In this paper, we show the predictability of the resource utilization by exploring the trends of recent cache usage. The time series based prediction is made with a machine learning approach and the prediction errors are small relative to the variation in the input data. This work would help understanding the characteristics of the resource utilization and plan for additional deployments of caches in the future.

1 Introduction

There has been a significant increase in data volume from various large scientific projects, such as the Large Hadron Collider (LHC) experiments. The High Energy Physics (HEP) community share their data generated from LHC with a world-wide community of users, which requires an increasingly larger volume of data to be transferred over the wide-area network. By 2028, the community expects the data volume to increase by thirty fold [1]. We observe that a significant portion of the popular datasets are shared among users in the same geographical region [1], which suggests that regional data storage caches could reduce data access latency by holding popular datasets closer to user analyses [2–9]. In-network cache

*e-mail: caitlinsim@berkeley.edu
**e-mail: kwu@lbl.gov
***e-mail: asim@lbl.gov
****e-mail: imonga@es.net
†e-mail: chin@es.net
‡e-mail: dhazen@es.net
§e-mail: fkw@ucsd.edu
¶e-mail: didavila@ucsd.edu
‖e-mail: newman@hep.caltech.edu
**e-mail: jbalcas@caltech.edu

or regional data caching mechanism [6–9] has been deployed in Southern California for the US CMS, one of the LHC experiment. The caching approach improves overall application performance by decreasing data access latency and increasing data access throughput. It also reduces traffic over the wide-area network by decreasing the number of repeated data transfers [10–12].

In this work, we examine the trends in data volume and cache utilization from the Southern California Petabyte Scale Cache (SoCal Cache) [6], which includes 23 federated caching nodes with approximately 2PB of total storage. From the trends, we also explored how much a machine learning model could predict the resource usage patterns. Our study shows that the number of data requests, cache hit rate, cache miss rate and so on could be reliably predicted a day ahead of time. This information could be used for short-term resource planning, such as network bandwidth reservation when heavy network traffic is expected the next day.

2 Background

Historically CMS analysis users would send their computing jobs to the sites where their input data was available. With the introduction of the Any Data, Anytime, Anywhere (AAA) service [13], also known as the "CMS Data Federation", certain jobs were allowed to read their data remotely. These jobs typically read a small percentage of the files they analyze and do so on stream mode, interleaving reading and processing, which allows them to tolerate some level of latency. Under this model of operation, computing resources and input data are no longer required to be in the same site, which makes the model more flexible at the cost of increased latency.

In order to hide the latency introduced by the remote reads, caches were added in the computing model. The technology used was born out of the XRootd framework, and commonly referred as XCache. XCaches were designed to deal with big files, big namespaces, and partial file reads. The Southern California Petabyte Scale Cache (SoCal Cache) is one of the pioneering ones. What makes SoCal Cache even more special is that it stretches over three different geographic zones: San Diego, Pasadena and Sunnyvale in California and serves two different CMS sites: UCSD and Caltech. Currently one server with 348TB of disk is dedicated to NANOAOD files, whilst the remaining CMS files could be cached on 22 servers with 1.6PB of disk. Table 1 shows the distribution of data servers.

Table 1: SoCal Cache distribution of data servers

location	# of servers	total disk space (TB)	data format
Pasadena (Caltech)	1	348	NANO
Pasadena (Caltech)	9	1327	MINI
San Diego (UCSD)	12	275	MINI
Sunnyvale (ESnet)	1	41	MINI

CMS organizes its data in a number of different data formats. The most common data formats used in analysis are AOD, MINIAOD and NANOAOD. Typically each file has the same number of events. Each MINIAOD file is much smaller than an AOD file, and similarly, a NANOAOD file is smaller than a MINIAOD file, as shown in Table 2 [14, 15]. Smaller data formats contain less details, and are not suitable for all analysis scenarios. It is nevertheless estimated that 50% of the analysis can be carried out using NANOAOD data [14]. Most analyses are conducted with MINIAOD files and NANOAOD files, and very occasionally, some users would access AOD files.

Table 2: Kilobytes per event on different CMS data formats

data format	kb per event
AOD	400-500
MINIAOD	40-50
NANOAOD	1-2

Table 3: Summary of data access from June 2020 to July 2023. About 68.6% of file requested are satisfied by this cache, and 62.8% bytes requested are in the cache.

	# of accesses	cache hit size (TB)	cache miss size (TB)	number of cache hits	number of cache misses
Total	27,315,865	19,877.91	11,771.45	18,736,392	8,579,473
Daily	23,629	17.20	10.18	16,207	7,421

Table 4: Summary of data access from September 2022 to July 2023. About 85.5% of the files requested and 94.0% of the bytes requested are cache hits.

	# of accesses	cache hit size (TB)	cache miss size (TB)	number of cache hits	number of cache misses
Total	5,889,264	10,824.09	690.75	5,038,749	850,515
Daily	18,233	33.41	2.14	15,599	2,633

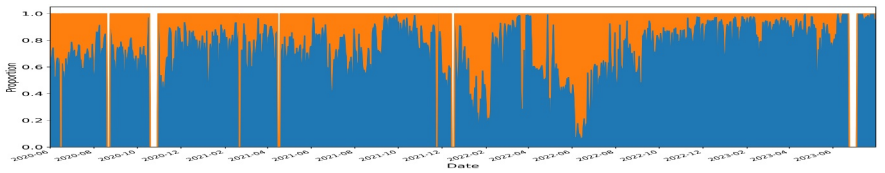
2.1 Monitoring

All cache servers are configured to send monitoring data via the default mechanism provided by the XRootD framework to a service called "The Shoveler". The Shoveler resides in UCSD, close and well-connected to the cache servers, and its function is to convert the unreliable UDP data into TCP and to transfer the data into a centralized RabbitMQ message bus operated by OSG. The data is later consumed by "The Collector", a home-made piece of software designed and operated by OSG, that assembles the different bits of data into fully-comprehensible access records and translates them into a json format. The json bits are then pushed into a StompMQ message bus managed by CERN and finally into an Elastic Search database. Every certain time these records are moved to a long-term storage based on HDFS from which we obtain the data for the analysis presented in this work.

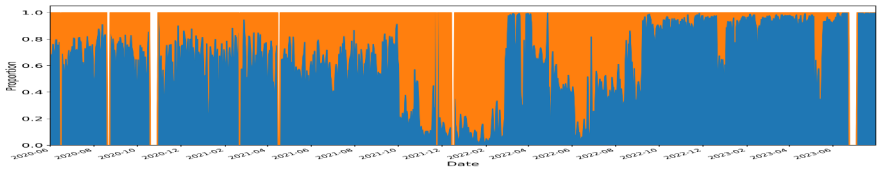
3 Cache utilization trends

Monitoring information collected from June 2020 to July 2023 in our study is stored in 8.8GB and 21,782 files. Table 3 shows the basic statistics about the data accesses during the study period. "Cache hit" represents when a data access request could be satisfied with a file in the cache, whereas "cache miss" means that a data transfer from a remote storage site is needed as the requested data file is not in the cache. About 68.6% of the 27 million data requests and 62.8% of the total 31PB data volume are cache hits. In recent 11 months from Sep. 2022 to July 2023, Table 4 shows that 85.5% of the requests and 94% of the requested bytes are cache hits.

Figure 1 shows the daily rates of cache hits (in blue) and cache misses (in orange). Figure 1a shows the rates based on the data request counts, and Figure 1b shows the proportion of the data volume. Figure 2 shows more information about the number of data requests and volume of requested data. Figure 2a shows the daily number of file requests, separating into cache hits (in blue) that could be satisfied with files in the cache and cache misses (in orange) that require wide-area data transfers. Figure 2b shows the daily volume of data requests.

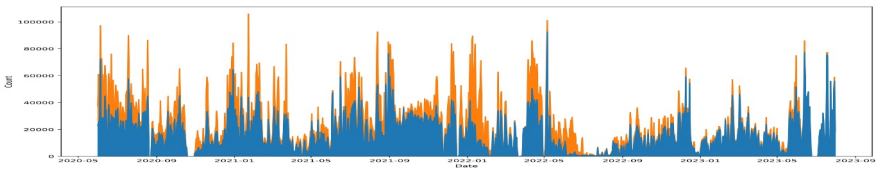


(a) Daily proportion of cache hit counts and cache miss counts. With 27.3 million total accesses, there are 18.73 million cache hits and 8.58 million cache misses. Overall, 68.6% of the total accesses has been satisfied by the cache.

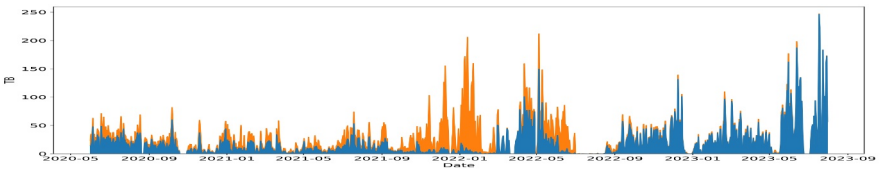


(b) Daily proportion of cache hit volume and cache miss volume. With 31.65PB of total bytes requested, there are 19.88PB served from the cache and 11.77PB from remote storage sites. Overall, 62.8% of the total traffic volume has been saved from the cache.

Figure 1: Daily proportion of cache hits (in blue) and cache misses (in orange) from June 2020 to July 2023.



(a) Daily counts of cache hits and cache misses. On an average day, there are 23,629 file requests, with 16,207 cache hits and 7,421 cache misses. The number accesses went over 100,000 on some days.



(b) Daily volume of cache hits and cache misses. On an average day, the total volume of files requested is 27.38TB with 17.20TB of cache hits and 10.18TB cache misses. The data volume went over 200TB on some days.

Figure 2: Daily accesses of cache hits (in blue) and cache misses (in orange) from June 2020 to July 2023.

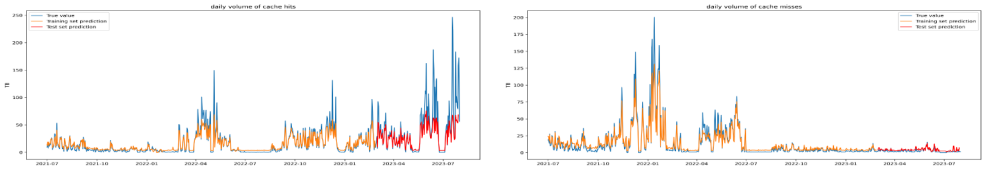
One noticeable pattern is shown between Oct. 2021 and Feb. 2022 in Figures 1b and 2b where cache miss is the dominant majority. This particular usage pattern [12, 16] is caused by a small number of users requesting larger AOD files that are not used very often. Since each AOD file is much larger than others, storing one of them in a disk cache could cause many smaller files (that are used more frequently) to be evicted. These large files are not accessed frequently, we could see the cache hit rate returns later. Between June 2023 and July 2023, the majority of requested data volume is cache hits. The cache hits ratio dominates approximately the same as from Sep. 2022 to July 2023, but the last two months have much higher data volume than the other periods.

4 Modeling the cache utilization

Our study on the cache utilization characterizes the trends of cache and network resource usage, next we'd like to see how additional caching nodes can be provisioned in the future. We built machine learning models with Long Short-Term Memory (LSTM) architecture [17, 18] for cache utilization trends for daily and hourly records. The results from the LSTM model [11, 12] are shown in Figures 3 and 4. Our model includes 5 features such as access counts, cache hit counts, cache miss counts, cache hit size and cache miss size. Our daily model has 608 records in the training set and 153 records in the testing set, over the study period from July 2021 to July 2023. The training data comes from the first 80% of the study period, and the testing data comes from the last 20%. Our hourly model has 6,412 records in the training dataset and 1,603 records in the testing dataset, over the study period from Sep. 2022 to July 2023. The daily model covers a longer period because the number of records are relatively smaller in a given period than the number of hourly records. In the remaining of this section, we mainly discuss the prediction of the data volume for the cache hits and misses.

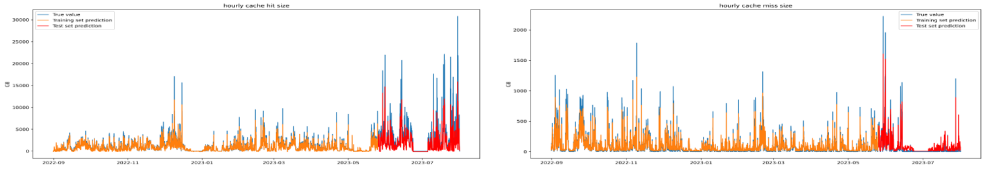
Table 5 shows the root-mean-square error (RMSE) of both the daily and hourly models for the data volume. The column labeled "standard deviation" is the standard deviation of the input data values. It provides a reference to determine how large the errors of predictions are. The relative ratio of testing RMSE and standard deviation are about 1σ, indicating the predictions are accurate enough for the models to be used in the resource provisioning.

Figures 3 and 4 show LSTM model results for the daily and hourly cache utilization in data volume. In these cases, we see the predictions in cache misses are closer to the actual values than the predictions for cache hits. The daily cache misses shows that the relative error in the testing dataset is 0.098, much less than others, where we observe stable usage trends. Figures 3a and 4a show significant number of spikes during the months of June and July of 2023 where we observe higher rates of cache hits. Large fluctuations are generally hard to predict. To demonstrate that this is true, we next study the predictions on the moving averages.



(a) (b)

Figure 3: Daily cache utilization from July 2021 to July 2023: (a) Daily volume of cache hits (b) Daily volume of cache misses



(a) (b)

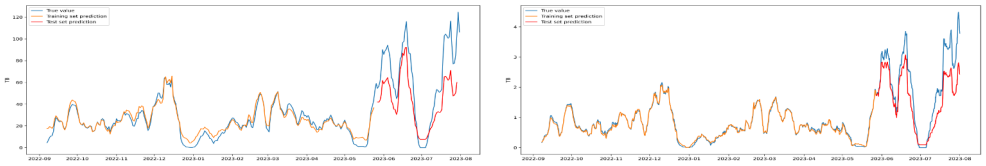
Figure 4: Hourly cache utilization from Sep. 2022 to July 2023: (a) Hourly volume of cache hits (b) Hourly volume of cache misses

Table 5: RMSE of daily/hourly LSTM model results for cache utilization. The relative prediction errors against the standard deviations are around 1.

	Training RMSE	Testing RMSE	standard deviation
Daily volume of cache hits	7.16	31.07	29.08
Hourly volume of cache hits	0.23	1.86	1.80
Daily volume of cache misses	6.91	2.26	22.93
Hourly volume of cache misses	0.09	0.42	0.36

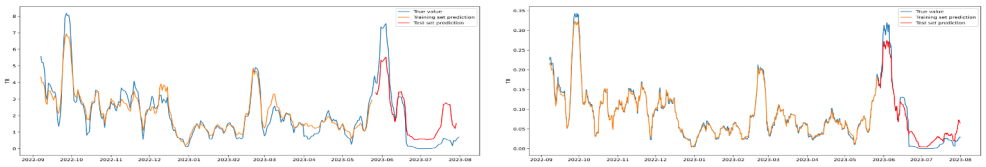
Figures 5 and 6 show the 7-day moving average of the daily and hourly volumes of cache hits and cache misses respectively for the same study period from Sep. 2022 to July 2023. We clearly see that the prediction results based on the moving-averages, shown in Figures 5 and 6, match better than the results based on the original time series data in Figures 3 and 4. Additionally, the hourly model in Figures 5b and 6b follow the 7-day moving averaged trends more closely than the daily models in Figure 5a and 6a during the same modeling period. The most likely reason might be there are more training data records for the hourly time series.

When the 7-day moving averaged daily model has more data records by extending the modeling periods from July 2021 to July 2023, both cache hits and cache misses follow the trends more closely in Figure 7a and Figure 7b than Figure 5a and Figure 6a respectively.



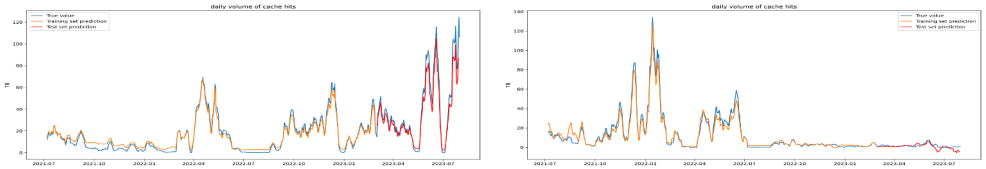
(a) (b)

Figure 5: 7-day moving average of the cache hits from Sep. 2022 to July 2023: (a) Daily volume of cache hits (b) Hourly volume of cache hits



(a) (b)

Figure 6: 7-day moving average of the cache misses from Sep. 2022 to July 2023: (a) Daily volume of cache misses (b) Hourly volume of cache misses



(a) (b)

Figure 7: 7-day moving average of the daily cache hits and daily cache misses from July 2021 to July 2023: (a) Daily volume of cache hits (b) Daily volume of cache misses

5 Discussion

From studying the logs from Southern California Petabyte Scale Cache, we observe a couple challenging issues. The first one is about the caching policy. As shown in Figures 1b and 2b, cache misses are high for some periods due to the cache pollution from accesses to a number of large files. These large files are used rarely and could displace a large number of smaller files due to their sizes. Adopting a separate caching policy for these larger files might improve effectiveness of the caching system. The second is about the monitoring. Figure 1 and 2 show some empty slots in the time periods where no data records are collected. During these time periods, the monitoring system appears to have stopped collecting information. Improving stability of the monitoring system would provide better information about the caching system and contribute to an early detection of the misbehavior or provide information for predicting cache system behavior.

Figures 3a and 4a show significant number of spikes for cache usage especially during the months of June and July in 2023. It would be important to understand the impact of these usage spikes in the longer term, especially for provisioning for deployment of future in-network caches. So far, we have only explored smoothing these spikes through moving averages as shown in Figures 5, 6 and 7. We would be interested in further study of these spikes.

6 Conclusion

SoCal Cache served on average about 62.8% of the requested data volume from its storage cache without remote transfers between June 2020 and July 2023, whereas the cache in recent 11 months from Sep. 2022 to July 2023 hits 94% of the requested data volume. The daily average volume of cache hits is about 17.2TB from June 2020 to July 2023 and about 33.4TB between Sep. 2022 and July 2023. We also explored the models for the cache utilization trends with a machine learning method known as LSTM, where the prediction errors (measured as RMSE) are small relative to the standard deviation of the input data.

This work shows the effectiveness of the caching system in decreasing data access latency, reducing wide-area network traffic, and consequently improving overall application performance. This work also helps understanding the characteristics of the resource utilization such as cache and network, and demonstrating the predictability of the resource utilization. We plan to study other caches currently under deployment to gain better understanding of the caching system, and explore longer-term predictability of the resource utilization and provisioning.

7 Acknowledgments

This work was supported by the Office of Advanced Scientific Computing Research, Office of Science, of the US Department of Energy under Contract No. DE-AC02-05CH11231, and also used resources of the National Energy Research Scientific Computing Center (NERSC). This work was also supported by the National Science Foundation through the grants OAC-1836650, PHY-2121686 and OAC-2112167.

References

[1] B. Brown, E. Dart, G. Rai, L. Rotman, J. Zurawski, *Nuclear physics network requirements review report*, University of California, Publication Management System Report LBNL-2001281, Energy Sciences Network (2020), `https://www.es.net/assets/Uploads/20200505-NP.pdf`

[2] D. Weitzel, M. Zvada, I. Vukotic, R. Gardner, B. Bockelman, M. Rynge, E. Hernandez, B. Lin, M. Selmeci, *StashCache: A Distributed Caching Federation for the Open Science Grid*, in *PEARC '19: Proceedings of the Practice and Experience in Advanced Research Computing on Rise of the Machines (learning)* (2019)

[3] L. Bauerdick, K. Bloom, B. Bockelman, D. Bradley, S. Dasu, J. Dost, I. Sfiligoi, A. Tadel, M. Tadel, F. Wuerthwein et al., *Xrootd, disk-based, caching proxy for optimization of data access, data placement and data replication*, Journal of Physics: Conference Series **513** (2014)

[4] X. Espinal, S. Jezequel, M. Schulz, A. Sciabà, I. Vukotic, F. Wuerthwein, *The quest to solve the hl-lhc data access puzzle*, EPJ Web of Conferences **245**, 04027 (2020)

[5] D. Weitzel, B. Bockelman, D.A. Brown, P. Couvares, F. Würthwein, E.F. Hernandez, *Data Access for LIGO on the OSG*, in *Proceedings of the Practice and Experience in Advanced Research Computing 2017 on Sustainability, Success and Impact* (2017)

[6] E. Fajardo, A. Tadel, M. Tadel, B. Steer, T. Martin, F. Würthwein, *A federated xrootd cache*, Journal of Physics: Conference Series **1085**, 032025 (2018)

[7] L. Bauerdick, D. Benjamin, K. Bloom, B. Bockelman, D. Bradley, S. Dasu, M. Ernst, R. Gardner, A. Hanushevsky, H. Ito et al., *Using xrootd to federate regional storage*, Journal of Physics: Conference Series **396**, 042009 (2012)

[8] E. Fajardo, D. Weitzel, M. Rynge, M. Zvada, J. Hicks, M. Selmeci, B. Lin, P. Paschos, B. Bockelman, A. Hanushevsky et al., *Creating a content delivery network for general science on the internet backbone using XCaches*, EPJ Web of Conferences **245**, 04041 (2020)

[9] Fajardo, Edgar, Tadel, Matevz, Balcas, Justas, Tadel, Alja, Würthwein, Frank, Davila, Diego, Guiang, Jonathan, Sfiligoi, Igor, *Moving the california distributed cms xcache from bare metal into containers using kubernetes*, EPJ Web of Conferences **245**, 04042 (2020)

[10] E. Copps, H. Zhang, A. Sim, K. Wu, I. Monga, C. Guok, F. Wurthwein, D. Davila, E. Fajardo, *Analyzing scientific data sharing patterns with in-network data caching*, in *4th ACM International Workshop on System and Network Telemetry and Analysis* (2021)

[11] R. Han, A. Sim, K. Wu, I. Monga, C. Guok, F. Wurthwein, D. Davila, J. Balcas, H. Newman, *Access Trends of In-network Cache for Scientific Data*, in *5th ACM International Workshop on System and Network Telemetry and Analysis* (2022)

[12] C. Sim, K. Wu, A. Sim, I. Monga, C. Guok, F. Wurthwein, D. Davila, H. Newman, J. Balcas, *Effectiveness and predictability of in-network storage cache for Scientific Workflows*, in *IEEE International Conference on Computing, Networking and Communication* (2023)

[13] K. Bloom, T. Boccali, B. Bockelman, D. Bradley, S. Dasu, J. Dost, F. Fanzago, I. Sfiligoi, A.M. Tadel, M. Tadel et al., *Any Data, Any Time, Anywhere: Global Data Access for Science*, in *IEEE/ACM 2nd International Symposium on Big Data Computing* (2015)

[14] A. Rizzi, G. Petrucciani, M. Peruzzi, *A further reduction in cms event data for analysis: the nanoaod format*, EPJ Web of Conferences **214**, 06021 (2019)

[15] *Miniaod analysis documentation* (2019), `https://twiki.cern.ch/twiki/bin/view/CMSPublic/WorkBookMiniAOD2017`

[16] T. Malik, R. Burns, A. Chaudhary, *Bypass caching: making scientific databases good network citizens*, in *21st International Conference on Data Engineering* (2005), pp. 94–105

[17] A. Sherstinsky, *Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network*, Physica D: Nonlinear Phenomena **404**, 132306 (2020)

[18] K. Greff, R.K. Srivastava, J. Koutník, B.R. Steunebrink, J. Schmidhuber, *LSTM: A search space odyssey*, in *IEEE transactions on neural networks and learning systems* (2016), Vol. 28, pp. 2222–2232