BenchCouncil Transactions on Benchmarks, Standards and Evaluations

TBench
Volume 2, Issue 4
2022

Research article

ISSN: 2772-4859

Review article

Short Communication

Reports

BenchCouncil Transactions on Benchmarks, Standards and Evaluations (TBench) is an open-access multi-disciplinary journal dedicated to benchmarks, standards, evaluations, optimizations, and data sets. This journal is a peer-reviewed, subsidized open access journal where The International Open Benchmark Council pays the OA fee. Authors do not have to pay any open access publication fee. However, at least one of the authors must register BenchCouncil International Symposium on Benchmarking, Measuring and Optimizing (Bench) (https://www.benchcouncil.org/bench/) and present their work. It seeks a fast-track publication with an average turnaround time of one month.

Contents

BenchCouncil Transactions on Benchmarks, Standards and Evaluations 2 (2022) 100083

Contents lists available at ScienceDirect

BenchCouncil Transactions on Benchmarks, Standards and Evaluations

journal homepage: www.keaipublishing.com/en/journals/benchcouncil-transactions-on-benchmarks-standards-and-evaluations/

Research Article

HPC AI500 V3.0: A scalable HPC AI benchmarking framework

Zihan Jiang [a,b,*], Chunjie Luo [a], Wanling Gao [a], Lei Wang [a], Jianfeng Zhan [a,b]

[a] Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
[b] University of Chinese Academy of Sciences, Beijing, China

ARTICLE INFO

Keywords:
Artificial intelligence
High performance computing
Benchmarking
Scalability

ABSTRACT

In recent years, the convergence of High Performance Computing (HPC) and artificial intelligence (AI) makes the community desperately need a benchmark to guide the design of next-generation scalable HPC AI systems. The success of the HPL benchmarks and the affiliated TOP500 ranking indicates that scalability is the fundamental requirement to evaluate HPC systems. However, being scalable in terms of these emerging AI workloads like deep learning (DL) raises nontrivial challenges. This paper formally and systematically analyzes the factor that limits scalability in DL workloads and presents HPC AI500 v3.0, a scalable HPC AI benchmarking framework. The HPC AI500 V3.0 methodology is inspired by bagging, which utilizes the collective wisdom of an ensemble of base models and enables the benchmarks to be adaptively scalable to different scales of HPC systems. We implement HPC AI500 V3.0 in a highly customizable manner, maintaining the space of various optimization from both system and algorithm levels. By reusing the representative workloads in HPC AI500 V2.0, we evaluate HPC AI500 V3.0 on typical HPC systems, and the results show it has near-linear scalability. Furthermore, based on the customizable design, we present a case study to perform a trade-off between AI model quality and its training speed. The source code of HPC AI500 V3.0 is publicly available from the HPC AI500 project homepage https://www.benchcouncil.org/aibench/hpcai500/.

1. Introduction

*D*eep *L*earning (DL) has been a dominating technology in *A*rtificial *I*ntelligence (AI) as its huge success in many challenging AI problems, such as image classification [1–3], object detection [4–6], and natural language processing [7–9]. DL allows building a computational model composed of multiple processing layers with trainable weights to learn the presentation of data [10]. To harness larger datasets and achieve higher model quality (e.g., Top1 accuracy), in recent years, tremendous DL models have been proposed endlessly, both for commercial applications [11–16] and scientific computing [17–20]. These giant models usually have deeper layers and billions of weights, which is extremely computation-intensive. Hence, academia and industry are greatly interested in designing and building next-generation HPC systems to run these emerging AI workloads for their computation requirement [21, 22]. Benchmark plays an important role in this process, as it provides the input and methodology for evaluation [23].

In the past three decades, the HPL benchmark [24] and the affiliated TOP500 ranking [25] witnessed the thriving of HPC systems. From CM-5 (1993) [26] to Fugaku (2020) [27], the FLOPS performance of the NO.1 supercomputer on the TOP500 list improves by more than $10^6\times$. HPL has become the measurement standard [28] in the HPC field for thirty years and will continue to be. The reason for its success is twofold. On the one hand, HPL solves a (random) dense linear system in double precision, which captures the general characteristic that many scientific applications share. We conclude this property as *relevancy*. On the other hand, HPL can adapt to scalable systems by adjusting the input matrix size. We summarize this property as *scalability*. The HPL lesson indicates that *relevancy* and *scalability* are two significant properties for an ideal benchmark. Most of the previous work [29–34] in AI benchmarking focus on relevancy and select represent workloads in real-world AI applications. However, they ignored the scalability issue.

Scalability is difficult to guarantee for AI workloads. According to the experiences in the previous researches [36,46], each AI workload has the best training batchsize, which is irrelevant to the system scale, to achieve state-of-the-art quality. This observation indicates that no matter how the scale of the system changes, the amount of parallel computation processed remains the same. Although many system optimizations [13,47–52] are proposed, all they can do is process this constant amount of computation as fast as possible by utilizing various parallel techniques (e.g., data parallelism [53]). Therefore, with the continuous growth of system scale, the speed of training existing AI workloads is rapidly accelerated. As shown in Fig. 1, from 2017 to 2021, with the development of HPC AI systems, the training time of

* Corresponding author at: Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China.
E-mail addresses: jiangzihan@ict.ac.cn (Z. Jiang), luochunjie@ict.ac.cn (C. Luo), gaowanling@ict.ac.cn (W. Gao), wanglei_2011@ict.ac.cn (L. Wang), zhanjianfeng@ict.ac.cn (J. Zhan).

https://doi.org/10.1016/j.tbench.2022.100083
Received 17 November 2022; Received in revised form 23 December 2022; Accepted 23 December 2022
Available online 29 December 2022

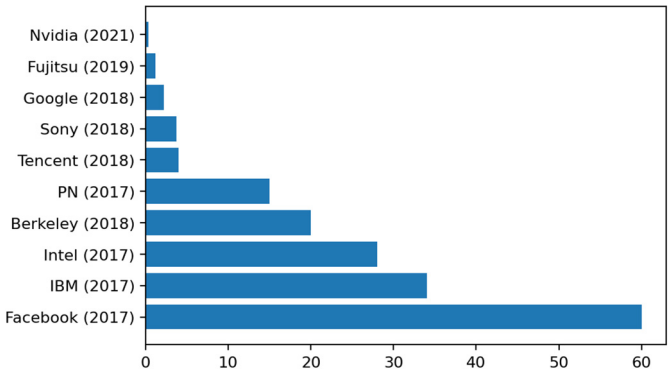


Fig. 1. ImageNet/ResNet-50 is a popular showcase for optimizing HPC AI systems from academia [35] and industry [36–44]. PN refers to Preferred Networks [45]. The x-axis refers to the training time measured in minutes.

ResNet-50 [2] has dropped exponentially, and the result of Nvidia [44] shows that it now can be done in under half a minute. From the benchmarking perspective, such a short running time does not allow for a thorough and endurable evaluation. Furthermore, the fixed amount of computation is distributed on the HPC system with a growing scale, which makes the resource utilization of each computing node extremely unsaturated.

Two prior works attempt to address the scalability problem in HPC AI benchmarking, namely AIPerf [54] and HPL-AI [55]. However, they both have their own flaws. AIPerf uses network architecture search (NAS) [56] as the primary workload. NAS automatically searches the network architecture with a predefined probability, introducing randomness to the benchmarking process. HPL-AI allows mixed-precision LU decomposition to solve a linear equation system and tends to be irrelevant to most AI workloads [57].

Bagging (Bootstrap Aggregation) [58] is designed to improve the stability and quality of the prediction by utilizing the collective wisdom of an ensemble of base models. As a meta-algorithm of ensemble learning [59], a critical feature of bagging is the independence between each base model. This independence makes bagging can be implemented as a highly parallel way to scale out with the number of nodes in an HPC system. Another merit of bagging is its flexibility and not being bound to any AI algorithm. In other words, we can easily and quickly achieve relevancy by integrating a state-of-the-art or state-of-the-practice algorithm into our bagging-based benchmarking framework. Considering the advantages above, this paper presents a bagging-based scalable AI benchmarking framework, which we call HPC AI500. HPC AI500 V3.0 extends our previous works: HPC AI500 V1.0 [50] and HPC AI500 V2.0 [57]. Table 1 summarizes the differences between HPC AI V3.0 from the other related works. HPC AI500 V3.0 not only leverages the advantages of bagging to achieve scalability and relevancy but also maintains user-customizable parallel optimization opportunities. HPC AI500 V3.0 implements two modules, bagging management (BM) and model parallelism management (MPM), to achieve this customizability. BM determines the algorithm adopted in data sampling and the number of base models. MPM determines the degree of parallelism inside each base model. Through these two modules, users can customize the number of base models and the degree of parallelism to make the trade-off between the model quality and training speed. Based on HPC AI500 [57], we evaluate HPC AI500 V3.0 on typical HPC systems to show its scalability and customizability.

Our main contributions are summarized as follows:

- According to the unique challenges of HPC AI Benchmarking, we reformulated the HPC AI scalability issue (Section 2).
- We propose the bagging approach in HPC AI benchmarking to achieve relevancy and scalability and implement HPC AI500 V3.0, a scalable and customizable framework for HPC AI benchmarking (Section 3).
- We evaluate HPC AI500 V3.0 by reusing HPC AI500 v2.0 workloads on typical HPC systems to show its scalability and customizability (Section 4).

2. Background and challenge

2.1. Deep learning preliminary

The whole training process of modern DL models is essentially a non-convex optimization. Mathematically, it can be represented as:

$$\min_{\forall x \in \mathbb{R}^n} f(x) := \frac{1}{N} \sum_{i=1}^{N} f_i(x), \tag{1}$$

where f_i is a loss function for data point $i \in \{1, 2, 3, \ldots, N\}$, which measures the deviation of the model prediction from the data. x is the vector of weights being optimized. The process of optimizing the loss function is called training and is performed iteratively.

2.1.1. Mini-batch stochastic gradient descent

Stochastic Gradient Descent (SGD) is the dominant method for training DL models. Vanilla SGD updates weight x by adding the gradient computed on a single data point of the whole dataset. Since only one random data point is processed at one iteration, this approach has two disadvantages. First, such a noisy update makes the training process unstable [62]. Second, the computation is inefficient, especially when using computing devices such as GPUs. Mini-batch SGD is proposed to remedy these two deficiencies. It minimizes the loss function f iteratively in the following form:

$$x_{k+1} = x_k - \eta_k \left(\frac{1}{|B_k|} \sum_{i \in B_k} \nabla f_i(x_k) \right) \tag{2}$$

where $B_k \in \{1, 2, 3, \ldots, N\}$ is the batch sampled from the whole dataset and η_k is the learning rate of iteration k. $|B_k|$ refers to the batchsize. The ratio of N and $|B_k|$ determines the number of iterations in a training epoch.

2.2. The scalability issue

With the convergence of AI and HPC, both academia and industry players [63–65] leverage the computing power of HPC systems to speed up the training process of DL models. However, SGD training has a significant drawback, limited by the batchsize.

2.2.1. The limitation of batchsize

Although there are millions of data in a DL dataset with the size N [66], the intrinsic sequential property of SGD only allows a batch with size B_k (e.g., $B_k = 256$) of data to be processed in parallel in an iteration. We call the computation cost required by a batch as the *Amount of Parallel Computation in an Iteration(in short, APC)*. Compared to Linpack, whose APC can be tuned by the size of the input matrix, the APC of DL workloads is usually a constant and can be represented as:

$$APC_{dl} = \sum_{j=1}^{|B_k|} Computation(f_j(x)) \tag{3}$$

where $j \in \{1, 2, 3, \ldots, |B_k|\}$ is data that is randomly sampled from the DL dataset with the size N and included in batch B_k. And $Computation(f_j(x))$ is the computation cost required by the DL model to process a single data and can be measured by FLOPs.

Eq. (3) indicates that APC_{dl} is determined by the $|B_k|$. However, the value of B_k is usually a small number, where $|B_k| \ll N$. Specifically, $B_k \in \{16, 32, 64, 256...512\}$ in many DL applications such as image classification [2] and object detection [4,5]. In this context, it is hard to fully utilize the computing power of HPC systems, which are usually equipped with hundreds or even thousands of nodes. Taking

Table 1
Comparison of HPC AI500 V3.0 against HPC AI500 V1.0, V2.0, and other HPC AI benchmarks. The equivalence, affordability, representativeness, and repeatability issues are resolved in our previous work HPC AI500 V2.0 [57]. HPC AI500 V3.0 is an HPC AI benchmarking framework which inherits and extends HPC AI500 V2.0 with scalability. HPC AI500 V3.0 can naturally integrate other HPC AI benchmarks. "✗" and "✓" indicate whether they have the corresponding properties. "-" indicates not verified.

Related work	Equivalence	Representativeness	Affordability	Repeatability	Scalability
HPC AI500 V1.0 (2018) [50]	✗	✓	✓	✗	✗
HPL-AI (2019) [55]	✓	✗	✓	✓	✓
Deep500 (2019) [60]	✗	✗	✓	–	✗
HPC AI500 V2.0 (2020) [57]	✓	✓	✓	✓	✗
AIPerf (2020) [54]	✓	✓	✓	–	✓
MLPerf (HPC) (2021) [61]	✓	✓	✓	✓	✗
HPC AI500 V3.0	✓	✓	✓	✓	✓

the ImageNet/ResNet-50 training on Summit [50,57] as an example. $|B_k| = 512$, according to Eq. (3), the APC of ImageNet/ResNet-50 is 11 776 GFLOPs. Considering Summit has 4608 nodes (six Nvidia Tesla GPUs in each node), each node only can allocate the computation of $\frac{11776}{4608} = 2.55$ GFLOPs, which is far away from the peak performance of six V100 GPUs.[1]

Naively enlarging $|B_k|$ to improve APC_{dl} leads to a degradation in the model quality due to the sharp minima [36,46,67]. The tricks proposed in [36,46,67] indeed increase $|B_k|$ to a larger number, but it is still far from the peak performance of the HPC system, leading to poor resource utilization. Furthermore, the proposed tricks are empirical, lack generalization ability, and depend on a specific DL workload. So far, no research can systematically and theoretically quantify the relationship between B_k and model quality.

2.2.2. The reformulation of HPC AI scalability

Based on the aforementioned analysis, we reformulate the HPC AI scalability from the following two perspectives. In the previous work [57], we have discussed how to resolve equivalence, representativeness, affordability, and repeatability issues.

- The APC_{dl} should be large enough to accommodate the scale and computing capability of HPC systems. To be specific, it is necessary to maintain a high resource utilization and near-linear speed up.
- The model quality should be maintained or improved while increasing the APC_{dl} and $|B_k|$. Otherwise, the whole training process is meaningless.

Compared to the traditional HPC scalability, which focuses on scale efficiency and resource utilization [24], the reformulated HPC AI scalability emphasizes the restraint of model quality and batchsize $|B_k|$.

2.3. Prior work

In addition to the other AI benchmarks [29–34], MLPerf (HPC) [61], HPL-AI [55], AIPerf [54], and HPC AI500 [50,57] are representative HPC AI benchmarking works. Among them, the earliest work is the HPC AI500 V1.0 [50], dating back to 2018. HPC AI500 V1.0 [50] and V2.0 [57] and MLPerf(HPC) fail to tackle the scalability issue and focus on selecting typical HPC AI applications and parallel-based optimizations. HPL-AI and AIPerf manage to achieve scalability but bring other problems. HPL-AI evaluates HPC systems by performing mixed-precision LU decomposition at the kernel level. Same to HPL, it can increase the APC by adjusting the size of the input matrix. However, LU decomposition is irrelevant to most AI workloads [57]. The AIPerf methodology is inspired by AutoML, whose core process is performed by NAS. Although AutoML can scale automatically with the number of nodes, the high randomness of NAS (Fig. 2) calls into question whether AutoML is desirable as an HPC AI benchmark. Table 1 summarizes the related work chronologically and compares our work with other related work in five dimensions.

[1] The peak performance of six V100 GPUs in terms of FLOPS is: $6 \times 15.7 \times 10^3$ GFLOPS $= 94.2 \times 10^3$ GFLOPS.

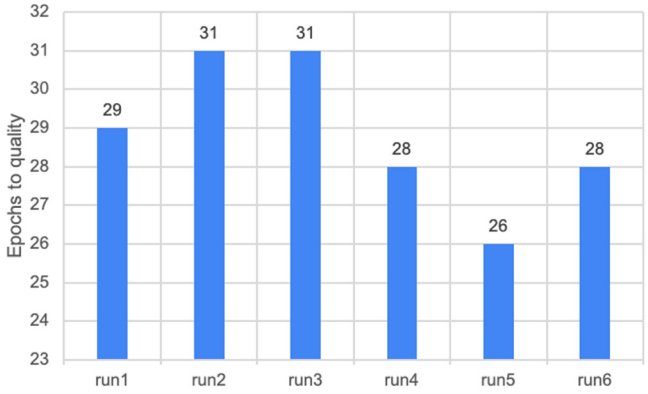


Fig. 2. The randomness of NAS. In different runs, the amount of computation required to train NAS to the target model quality varies, which leads to unfair and unrepeatable evaluation.

3. HPC AI500 V3.0

This section first presents the HPC AI500 v3.0 methodology. Then we detail the design, workflow, and customizable configuration. Finally, we introduce the measurement method and the proposed metrics.

3.1. Methodology

3.1.1. Ensemble learning and bagging

The ensemble learning idea is to solve a common problem by combining the predictions of a group of base models. Rather than making decisions depending on a single model, a group of models makes it possible for ensemble learning to reduce the variance of predictions [59], so-called the wisdom of crowds [68]. Bagging (Bootstrap AGGregatING) is a fundamental paradigm of Ensemble learning. As its name suggests, bagging consists of two parts: bootstrapping and aggregating. Bootstrapping is essentially a data sampling process with replacement from the original dataset. The data generated through this process is called the bootstrapped dataset. The training process of bagging is highly parallel as each base model in the ensemble is trained based on its corresponding bootstrapped dataset rather than the original dataset. After finishing the training, the final decision is aggregated by averaging all the predictions of the base models.

3.1.2. Applying bagging in HPC AI benchmarking

For HPC AI benchmarking, to tackle the scalability problem, the first thing is to enlarge the APC to keep up with the increasingly larger scale of HPC systems. Inspired by the Bagging, we introduce *the base model ensemble* on the basis of the training of a single model in the previous AI benchmark like HPC AI500 V2.0. We rewrite Eq. (3) in the following bagging form:

$$APC_{dl} = \sum_{m=1}^{M} \sum_{j=1}^{|B_k|} Computation(f_{m,j}(x)) \tag{4}$$

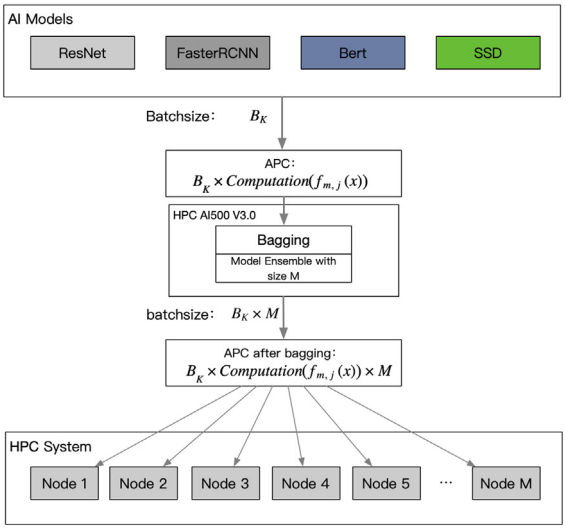


Fig. 3. The system overview of HPC AI500 V3.0. APC refers to the amount of parallel computing in an iteration.

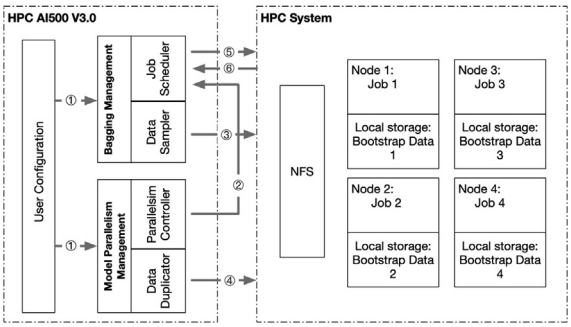


Fig. 4. System design and workflow of HPC AI500 V3.0. NFS refers to the Network File System of HPC systems that each node shares.

where M is the number of the base models in the ensemble, f_m is the m_{th} base model. Note that each base model is the instance of the original model, so the computation cost of each base model is equivalent to that in Eq. (3). Compared to AutoML, the re-sampled bootstrapped dataset makes every base model dissimilar, but the computational logic of each model is consistent, guaranteeing no randomness shown in Fig. 2. All the base model in the ensemble is trained independently, enlarging the $|B_k|$ by M times, and so does APC_{dl}. Considering each base model may train in a distributed manner across several nodes, the ensemble size M and the parallelism degree inside a base model P_degree should satisfy Eq. (5), where Sys_{scale} refers to how many nodes are contained in an HPC system.

$$Sys\_scale = M \times P\_degree \tag{5}$$

3.2. System overview

Based on the Bagging approach, we present HPC AI500 V3.0 and the system overview shown in Fig. 3. HPC AI500 V3.0 does not focus on workload selection and construction as previous AI benchmarks [29,31,34]. Instead, it is a framework that is compatible with these efforts. We briefly introduce the positioning and role of HPC AI500 V3.0 through Fig. 3. This figure shows that HPC AI500 V3.0 scales out the upper-layer AI workloads on lower-layer HPC systems by adaptively increasing APC_{AI}. Specifically, the batchsize of each AI workload is initially only a fixed B_k. After Bagging, a set of M base models are generated, which increases APC_{dl} by concurrently running M base models. This way, thereby, achieves higher resource utilization. In addition, the size of the base model set, M, can be adjusted according to the system size, corresponding to the same adjustable input matrix size in HPL, to adapt to the future growth of the HPC system scale.

3.3. System design and workflow

HPC AI500 V3.0 consists of three components, namely, User Configuration (UC), Bagging Management (BM), and Model Parallelism Management (MPM). BM focuses on managing Bagging, including Job Controller and Data Sampler. Job Controller schedules M jobs to the corresponding nodes, then launch training, and finally aggregates the predictions. Note that each job corresponds with a base model training. Data Sampler controls the data sampling algorithm. MPM is divided into Parallelism Controller and Data Duplicator. Parallelism Controller sets the parallel mode and P_degree. Data Duplicator is responsible for copying and migrating data according to parallelism-related configuration. As shown in Fig. 4, we summarize the workflow of HPC AI500 V3.0 as follows:

1. UC sends the configurations to BM and MPM. BM receives the configurations, including job number, equal to ensemble size M, and saves the DL model and original dataset that needs to be trained. MPM receives the configurations, such as parallelism mode, P_{degree}, and Sys_{scale}.
2. Parallelism Controller in MPM checks if M, P_{degree}, and Sys_{scale} satisfy Eq. (5) and generates the mapping of the jobs to the nodes according to the received messages (e.g., $Task1 \rightarrow Node1$), then sends this mapping to Job Scheduler in BM.
3. Data Sampler in BM determines the sampling algorithm and generates the bootstrap data for each task. All the generated data is sent to the NFS of the HPC system.
4. Data Duplicator in MPM duplicates the bootstrap data according to the mapping that Parallelism Controller generates. For example, $Job1 -> Node1$ means the bootstrap data in Job1 only need

Table 2
The Customizable Configuration of HPC AI500 V3.0. $Node\_acc$ refers to the number of accelerators equipped in a node of the HPC system.

Type	Default setting	Alternatives
Basic	$P_{degree} = Node\_acc$ $M = \frac{Sys_{scale}}{P_{degree}}$	Any M and P_{degree} that satisfy Eq. (5)
Learning Rate Scheduler	warm-up schema and linear scaling [69]	LARS [35], LAMB [70]
Optimizer	SGD with momentum	Adam [71], AdaGrad [72]
Data Precision for Training	FP16	mixed-precision, Int8
Data Precision for Communication	FP32	FP16, Int8
Parallel Mode	data parallelism	model parallelism, pipeline parallelism [73], mixed parallelism
Communication Mode	synchronous all-reduce	2D-Torus [41], Hierarchical all-reduce [40]
Framework	TensorFlow [74]	PyTorch [75], Mindspore [76]

to be duplicated once. All the duplicated data is sent to the local storage of the corresponding node.

5. Job Scheduler sends the job to the corresponding nodes and launches the training of the whole ensemble.
6. After the training is finished, Job Scheduler collects all the ensemble output and then makes the final prediction.

3.4. Customizable configuration

In order to maintain the optimization space, in addition to the basic configuration, such as M and $|P_{degree}|$, we summarize other customizable configurations in Table 2. We provide a default setting and some alternatives in each configuration type. Note that alternatives just list the favored option, and the user can customize the efficient implementation according to their situation.

3.5. Metrics

Same as HPL, we use *FLOPS* (Floating point operations per second) as our primary metric:

$$FLOPS = \frac{\sum_{i=1}^{N/|B_k|} APC_{dl}}{T_{epoch}} \tag{6}$$

where T_{epoch} refers to the training time of one epoch and $N/|B_k|$ refers to the number of iterations in one training epoch. In addition to FLOPS, we also adopt a metric that considers both system throughput and model quality, namely Valid FLOPS (VFLOPS) [57]. The definition of *VFLOPS* is shown as follows:

$$VFLOPS = FLOPS * penalty\_coefficient \tag{7}$$

$$penalty\_coefficient = (achieved\_quality/target\_quality)^n \tag{8}$$

where $penalty\_coefficient$ is used to penalize or award the FLOPS based on the achieved quality. $achieved\_quality$ refers to the actual model quality achieved in the evaluation. $target\_quality$ is predefined in the Table 4. The value of n defines the sensitivity to the model quality. According to the setting of HPC AI500 V2.0 [57], we set n as 10 for Extreme Weather Analytics and 5 for Image Classification.

Table 3
The FLOPs calculation rules for primary operators in a DL model. K refers to the kernel size, C_{in} and C_{out} refers to the input and output channel, H and W refers to the data size, $Group_{size}$ refers to the group size of the convolution, and FL refers to the flatten layer used in the Fully-connected.

Operators	FLOPs
Convolution	$2 \times K^2 \times C_{in} \times H \times W \times C_{out}$
Depth-wise Convolution	$2 \times K^2 \times C_{in} \times H \times W$
Group Convolution	$\frac{2 \times K^2 \times C_{in} \times H \times W \times C_{out}}{Group_{size}}$
Fully-connected	$FL_{in} \times FL_{out}$
Element-wise	$C_{out} \times H \times W$
Pooling	$C_{in} \times H \times W$
Normalization	$C_{in} \times H \times W$

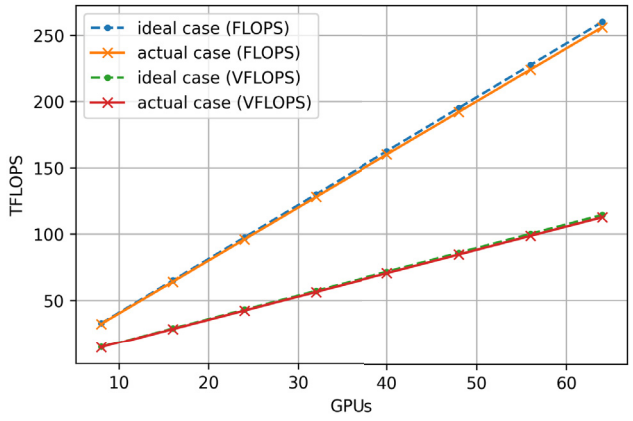
3.6. Measurement

According to Eq. (4) and Eq. (6), to determine the $FLOPS$, we need to first measure the $Computation(f(x))$. Although profiling tools such as Nsight [77] are able to count the FLOPs by kernel replay, it is dependent on the Nvidia hardware. In order to reduce the influence of the hardware and the hardware-specific optimizations performed by bundled low-level libraries (e.g., CuDnn for Nvidia GPUs), we present an analytical method to calculate the FLOPs that a DL model requires.

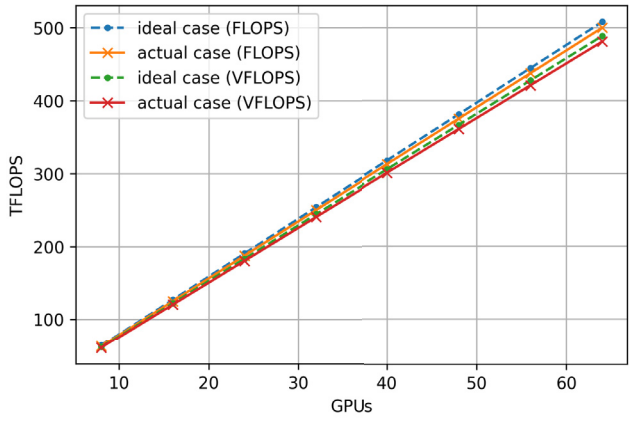
Modern AI frameworks, such as TensorFlow, describe the computation of a DL model using a directed acyclic graph (DAG) that consists of multiple nodes and edges. The Node in the DAG represents a kind of operator, and the edge represents the data flow. Each operator defines a computation logic and receives the data from the input edge, and then sends the intermediate result to the next operator after finishing its computation. Unlike HPL, which has only one kind of operator (LU decomposition), a DL model usually consists of multiple operators with different kinds. Hence, we summarize the most frequent operators in DL as shown in Table 3. In addition to these listed operators, we ignore other low-proportion operators contained in the DL model. Based on this table, we can calculate the $Computation(f(x))$ by traversing the DAG.

3.7. Implementation details

Job scheduler of the Bagging management module is based on SLURM (Simple Linux Utility for Resource Management) [78]. SLRUM is the most commonly used scheduling system in HPC AI systems, fault-tolerant and highly scalable, and suitable for Linux clusters of different sizes. We implement the submitted job script based on the $sbatch$ interface of SLRUM and use $sinfo$ and $smap$ to monitor the training progress of the base model in each job, and the basic unit of job scheduling is a container implemented by Docker [79]. According to the literature [58], the implemented random sampling algorithm guarantees that the i_{th} training sample is selected n ($n \in \{0, 1, 2...\}$) times. The probability of the times approximates the Poisson distribution of $\lambda = 1$, so the probability of at least one occurrence of the i_{th} sample is $1 - (\frac{1}{e}) = 0.632$. So for any Bagging base classifier, about 36.8% of the samples of the original dataset will not be used at the time of training. The default parallel implementation in the parallel management module uses data parallelism implemented by Horovod and OpenMPI, which is also the most common parallel method in HPC AI systems [17–19]. The measurement of bandwidth is divided into intra-node communication and inter-node communication, and we use Nvidia-smi (NVIDIA System Management Interface) tool [80] to monitor communication within nodes and use iftop tool [81] to monitor communication between nodes.



(a) Extreme Weather Analytics.



(b) Image Classification.

Fig. 5. The scalability experiments of HPC AI500 V3.0 in terms of FLOPS and VFLOPS. The $penalty\_coef\,ficient$ is 0.44 for Extreme Weather Analytics and 0.96 for Image Classification.

4. Evaluation

4.1. Experimental setup

4.1.1. Hardware

Our experiments are conducted on a 64GPUs-cluster, consisting of eight nodes, each of which is equipped with one Intel(R) Xeon(R) Platinum 8268 CPU and eight NVIDIA Tesla V100 GPUs. Each GPU in the same node has 32 GB HBM memory, connected by NVIDIA NVLink—a high-speed GPU interconnection whose theoretical peak bi-directional bandwidth is 300 GB/s. The nodes are connected with Ethernet networking with a bandwidth of 10 Gb/s. Each node has 1.5 TB system memory and an 8 TB NVMe SSD disk.

4.1.2. Software

We use TensorFlow v1.14, compiled with CUDA v10.1 and cuDnn v7.6.2 backend. We use Horovod v0.16.4 for synchronous distributed training, compiled with OpenMPI v3.1.4 and NCCL v2.4.8. NCCL is short for the NVIDIA Collective Communications Library, which is a closed-source library of multi-GPU collective communication primitives that are topology-aware.

4.2. Workloads

HPC AI500 V3.0 is a benchmarking framework, which means any AI benchmark can be integrated into this framework in a bagging manner. Here, our default implementation is based on HPC AI500 V2.0 [57], a well-received HPC AI benchmark that mainly consists of two workloads, covering AI applications in business and scientific computing. As shown in Table 4, Image Classification uses ResNet-50 [2]and ImageNet [66] for training, which is a well-known showcase for optimizing HPC AI systems. Extreme Weather Analytics [82] is a representative scientific application, it uses Faster-RCNN for detecting the extreme weather in the climate image. Each climate image in Extreme Weather Dataset consists of 16 channels and contains four extreme weather patterns.

4.3. The scalability experiments

The scalability experiments are conducted with the default setting of HPC AI500 V3.0, as shown in Table 2. We set the $P_degree = 8$, which is equal to the number of GPUs in a node. In each node, a job is distributed to 8 GPUs by using data parallelism. We perform the experiment sequentially on different system scales, typically the $Sys_scale = 8, 16, 24, 32, 40, 48, 56, 64$ GPUs. According to Eq. (5), the corresponding job number is $M = 1, 2, 3, 4, 5, 6, 7, 8$. The results of scalability experiments are shown in Fig. 5. As we can see, HPC AI500 V3.0 shows near-linear scalability in both FLOPS and VFLOPS. Note that, in Fig. 5(a), the $penalty\_coef\,ficient = 0.44$ leads to a gap between the VFLOPS line and FLOPS of Extreme Weather Analytics. Furthermore, we measure the GPU utilization by Nsight at the scale of 64 GPUs and the result is shown in Fig. 6. Both Extreme Weather Analytics and Image Classification achieve high GPU utilization.

4.4. The customizability experiments

4.4.1. Trade-off between the model quality and training speed

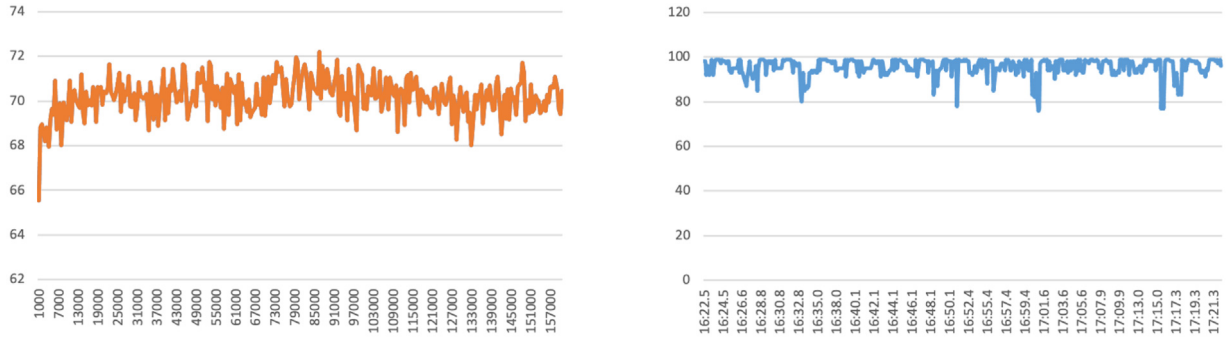
To exhibit this trade-off, we take Image Classification as the showcase. We set the $M = 8, 4, 1$ while the corresponding $P_{degree} = 8, 16, 64$. As shown in Fig. 7, the training speed increases along with a decrease in M. When $M = 1$, the process becomes training a single model through the whole cluster, achieving the highest training speed. However, since only one model makes decisions in the ensemble, the model quality suffers about a 3% drop compared to the case of $M = 8$. In practical scenarios, users can choose appropriate M and P_{degree} according to their training speed and model quality requirements.

4.4.2. Optimizations

To show the customizability of HPC AI500, we implement two frequently-used optimizations, mixed-precision training, and communication compression. The former utilizes Tensor Cores in Nvidia Volta architecture to accelerate the model's fully-connected and convolution layer, allowing a fused-multiply-add computation. When performing mixed precision training with a Tensor Core, we use FP16 for calculation and FP32 for accumulation. The latter is the communication compress-on that compresses the tensor precision for synchronizing from 32FP to 16FP to reduce communication overhead. We configure the optimization experiments in the same way as Section 4.3, and the results are shown in Fig. 8. We compared the optimized version to the original version to observe the corresponding effect. Since mixed-precision Extreme Weather Analysis leads to a significant loss of the model quality, here we only report the performance of the model compression. As we can see, mixed-precision training brings about 2x speed up for Image Classification. As for communication compression, it brings about 1.2x for Extreme Weather Analytics but barely has any speed up on Image Classification. The size of the communication tensor in Extreme Weather Analytics is 1.6x larger than that of Image Classification, allowing Extreme Weather Analytics to get a notable benefit.

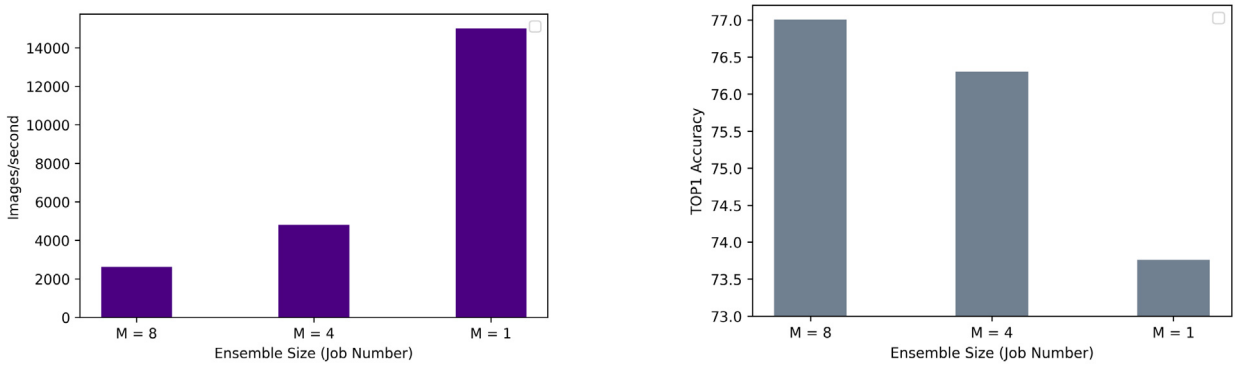
Table 4
The Specification of HPC AI500 V2.0 workloads [57]. HPC AI500 V3.0 can integrate any HPC AI benchmarks. In our evaluation, we reuse the HPC AI500 V2.0 workloads for testing.

Problem domains	Models	Datasets	Target quality
Image Classification	ResNet-50	ImageNet	TOP1 Accuracy = 0.763
Extreme Weather Analytics	Faster-RCNN	Extreme Weather Dataset	mAP@[IoU=0.5] = 0.35



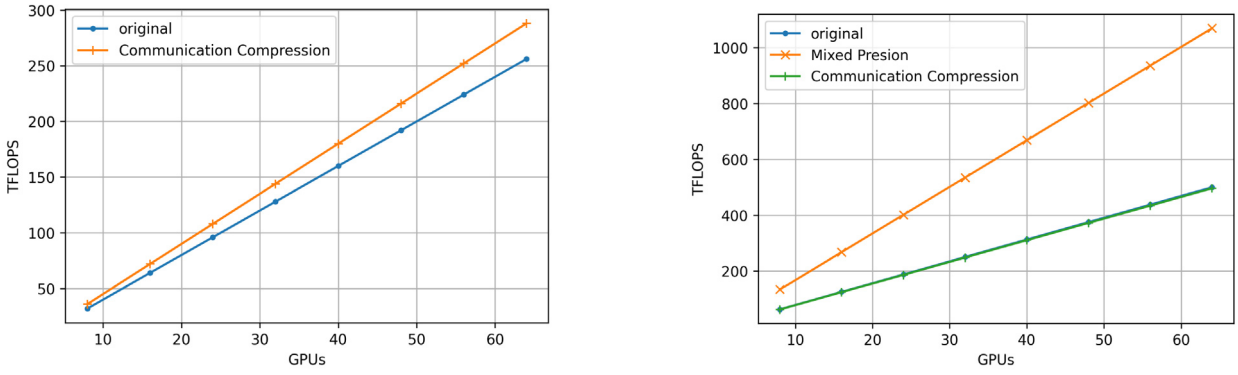
(a) Extreme Weather Analytics. **(b)** Image Classification.

Fig. 6. GPU utilization (%) of HPC AI500 V3.0. The X-axis represents different time steps.



(a) The training speed of different configurations. **(b)** The model quality of different configurations.

Fig. 7. The trade-off between the training speed and model quality. The workload is Image Classification. We use images per second to indicate how fast the training is.

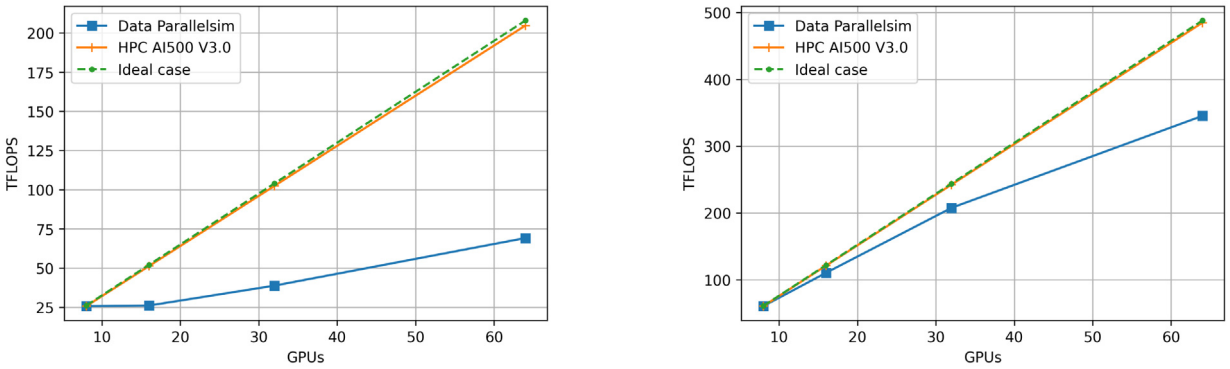


(a) Extreme Weather Analytics. **(b)** Image Classification.

Fig. 8. The optimization experiments of HPC AI500 V3.0. In Fig. 8(b), the lines of the original and mixed precision overlap for their similar performance.

4.5. Comparison experiments

We compare our work with data parallelism (DP), which is a mainstream parallel method used in many previous work [17,18,47,61]. In this experiment, we focus on scale efficiency in terms of FLOPS. The system scales from 8 GPUs to 64 GPUs. As shown in Fig. 9, the scaling efficiency of DP is much lower than our approach in both Extreme Weather Analysis and Image Classification. The heavy communication



(a) Extreme Weather Analytics. **(b)** Image Classification.

Fig. 9. The comparison experiments between HPC AI500 V3.0 against a setting using data parallelism.

overhead of DP is the main reason for this phenomenon because all the model copies of DP need to be synchronized globally at the end of each training step. The base model in the model ensemble of HPC AI500 V3.0 is trained highly independently without synchronization, so the communication overhead is avoided.

5. Conclusion

In this paper, we reformulate the HPC AI scalability issue and present HPC AI500 V3.0, a scalable and customizable framework for HPC AI benchmarking. The methodology of HPC AI500 V3.0 allows users to integrate existing AI benchmarks in a bagging manner, a meta-algorithm of ensemble learning with intrinsic high parallelism, leading to scalable benchmarking. The bagging management and model parallelism management of HPC AI500 V3.0 gives users the flexibility to control the size of model ensembles and the degree of model parallelism, enabling various optimizations from both system and algorithm levels. Based on HPC AI500 V2.0, which tackles the equivalence, representativeness, affordability, and repeatability issues, HPC AI500 V3.0 provide a complete HPC AI benchmarking framework. Reusing the workloads of HPC AI500 V2.0, we evaluate HPC AI500 V3.0 on a typical HPC system and the experimental results show the scalability and customizability of the proposed benchmarking framework.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, Adv. Neural Inf. Process. Syst. 25 (2012).
[2] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.
[3] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, Rethinking the inception architecture for computer vision, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 2818–2826.
[4] S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: Towards real-time object detection with region proposal networks, Adv. Neural Inf. Process. Syst. 28 (2015).
[5] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, A.C. Berg, SSD: Single shot multibox detector, in: European Conference on Computer Vision, Springer, 2016, pp. 21–37.
[6] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 779–788.
[7] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, Adv. Neural Inf. Process. Syst. 30 (2017).
[8] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, 2018, arXiv preprint arXiv:1810.04805.
[9] K. Han, A. Xiao, E. Wu, J. Guo, C. Xu, Y. Wang, Transformer in transformer, Adv. Neural Inf. Process. Syst. 34 (2021).
[10] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (7553) (2015) 436–444.
[11] OpenAI, OpenAI: AI and Compute, https://openai.com/blog/ai-and-compute/.
[12] A. Gholami, Medium: AI and Memory Wall, https://medium.com/riselab/ai-and-memory-wall-2cb4265cb0b8/.
[13] M. Shoeybi, M. Patwary, R. Puri, P. LeGresley, J. Casper, B. Catanzaro, Megatron-LM: Training multi-billion parameter language models using model parallelism, 2019, arXiv preprint arXiv:1909.08053.
[14] T. Brown, B. Mann, N. Ryder, M. Subbiah, J.D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al., Language models are few-shot learners, Adv. Neural Inf. Process. Syst. 33 (2020) 1877–1901.
[15] D. Lepikhin, H. Lee, Y. Xu, D. Chen, O. Firat, Y. Huang, M. Krikun, N. Shazeer, Z. Chen, Gshard: Scaling giant models with conditional computation and automatic sharding, 2020, arXiv preprint arXiv:2006.16668.
[16] W. Fedus, B. Zoph, N. Shazeer, Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity, 2021, arXiv preprint arXiv:2101.03961.
[17] A. Mathuriya, D. Bard, P. Mendygral, L. Meadows, J. Arnemann, L. Shao, S. He, T. Kärnä, D. Moise, S.J. Pennycook, et al., CosmoFlow: Using deep learning to learn the universe at scale, in: SC18: International Conference for High Performance Computing, Networking, Storage and Analysis, IEEE, 2018, pp. 819–829.
[18] T. Kurth, S. Treichler, J. Romero, M. Mudigonda, N. Luehr, E. Phillips, A. Mahesh, M. Matheson, J. Deslippe, M. Fatica, et al., Exascale deep learning for climate analytics, in: SC18: International Conference for High Performance Computing, Networking, Storage and Analysis, IEEE, 2018, pp. 649–660.
[19] W. Jia, H. Wang, M. Chen, D. Lu, L. Lin, R. Car, E. Weinan, L. Zhang, Pushing the limit of molecular dynamics with ab initio accuracy to 100 million atoms with machine learning, in: SC20: International Conference for High Performance Computing, Networking, Storage and Analysis, IEEE, 2020, pp. 1–14.
[20] Z. Guo, D. Lu, Y. Yan, S. Hu, R. Liu, G. Tan, N. Sun, W. Jiang, L. Liu, Y. Chen, et al., Extending the limit of molecular dynamics with ab initio accuracy to 10 billion atoms, 2022, arXiv preprint arXiv:2201.01446.
[21] Oak Ridge National Laboratory, Summit, https://www.olcf.ornl.gov/summit/.
[22] Fujitsu, Fugaku, https://www.fujitsu.com/global/about/innovation/fugaku/.
[23] J.L. Hennessy, D.A. Patterson, Computer Architecture: A Quantitative Approach, Elsevier, 2011.
[24] J.J. Dongarra, P. Luszczek, A. Petitet, The LINPACK benchmark: Past, present and future, Concurr. Comput.: Pract. Exper. 15 (9) (2003) 803–820.
[25] J. Dongarra, Top500 Website, https://www.top500.org/.
[26] J. Dongarra, CM-5 in TOP500 List, https://www.top500.org/lists/top500/1993/06/.
[27] J. Dongarra, Fugaku in TOP500 List, https://www.top500.org/news/japan-captures-top500-crown-arm-powered-supercomputer/.
[28] J. Zhan, Call for establishing benchmark science and engineering, 2021, arXiv preprint arXiv:2112.09514.
[29] R. Adolf, S. Rama, B. Reagen, G.-Y. Wei, D. Brooks, Fathom: Reference workloads for modern deep learning methods, in: 2016 IEEE International Symposium on Workload Characterization, IISWC, IEEE, 2016, pp. 1–10.
[30] C. Coleman, D. Narayanan, D. Kang, T. Zhao, J. Zhang, L. Nardi, P. Bailis, K. Olukotun, C. Ré, M. Zaharia, Dawnbench: An end-to-end deep learning benchmark and competition, Training 100 (101) (2017) 102.
[31] H. Zhu, M. Akrout, B. Zheng, A. Pelegris, A. Phanishayee, B. Schroeder, G. Pekhimenko, TBD: Benchmarking and analyzing deep neural network training, 2018, arXiv preprint arXiv:1803.06905.

[32] W. Gao, F. Tang, L. Wang, J. Zhan, C. Lan, C. Luo, Y. Huang, C. Zheng, J. Dai, Z. Cao, et al., AIBench: An industry standard internet service AI benchmark suite, 2019, arXiv preprint arXiv:1908.08998.
[33] V.J. Reddi, C. Cheng, D. Kanter, P. Mattson, G. Schmuelling, C.-J. Wu, B. Anderson, M. Breughe, M. Charlebois, W. Chou, et al., Mlperf inference benchmark, in: 2020 ACM/IEEE 47th Annual International Symposium on Computer Architecture, ISCA, IEEE, 2020, pp. 446–459.
[34] P. Mattson, C. Cheng, G. Diamos, C. Coleman, P. Micikevicius, D. Patterson, H. Tang, G.-Y. Wei, P. Bailis, V. Bittorf, et al., Mlperf training benchmark, Proc. Mach. Learn. Syst. 2 (2020) 336–349.
[35] Y. You, Z. Zhang, C.-J. Hsieh, J. Demmel, K. Keutzer, Imagenet training in minutes, in: Proceedings of the 47th International Conference on Parallel Processing, 2018, pp. 1–10.
[36] P. Goyal, P. Dollár, R. Girshick, P. Noordhuis, L. Wesolowski, A. Kyrola, A. Tulloch, Y. Jia, K. He, Accurate, large minibatch SGD: Training imagenet in 1 hour, 2017, arXiv preprint arXiv:1706.02677.
[37] T. Akiba, S. Suzuki, K. Fukuda, Extremely large minibatch SGD: Training resnet-50 on imagenet in 15 minutes, 2017, arXiv preprint arXiv:1711.04325.
[38] M. Cho, U. Finkler, S. Kumar, D. Kung, V. Saxena, D. Sreedhar, Powerai DDL, 2017, arXiv preprint arXiv:1708.02188.
[39] V. Codreanu, D. Podareanu, V. Saletore, Scale out for large minibatch SGD: Residual network training on ImageNet-1K with improved accuracy and reduced time to train, 2017, arXiv preprint arXiv:1711.04291.
[40] X. Jia, S. Song, W. He, Y. Wang, H. Rong, F. Zhou, L. Xie, Z. Guo, Y. Yang, L. Yu, et al., Highly scalable deep learning training system with mixed-precision: Training imagenet in four minutes, 2018, arXiv preprint arXiv:1807.11205.
[41] H. Mikami, et al., Imagenet/resnet-50 training in 224 seconds, 2018, arXiv preprint arXiv:1811.05233.
[42] C. Ying, S. Kumar, D. Chen, T. Wang, Y. Cheng, Image classification at supercomputer scale, 2018, arXiv preprint arXiv:1811.06992.
[43] M. Yamazaki, A. Kasagi, A. Tabuchi, T. Honda, M. Miwa, N. Fukumoto, T. Tabaru, A. Ike, K. Nakashima, Yet another accelerated SGD: Resnet-50 training on imagenet in 74.7 seconds, 2019, arXiv preprint arXiv:1903.12650.
[44] MLCommons, MLPerf-Training-Result-V1.1, https://mlcommons.org/en/training-normal-11//.
[45] Preferred networks website, https://www.preferred.jp/en/.
[46] N.S. Keskar, D. Mudigere, J. Nocedal, M. Smelyanskiy, P.T.P. Tang, On large-batch training for deep learning: Generalization gap and sharp minima, 2016, arXiv preprint arXiv:1609.04836.
[47] A. Sergeev, M. Del Balso, Horovod: Fast and easy distributed deep learning in TensorFlow, 2018, arXiv preprint arXiv:1802.05799.
[48] J. Rasley, S. Rajbhandari, O. Ruwase, Y. He, Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters, in: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2020, pp. 3505–3506.
[49] N. Shazeer, Y. Cheng, N. Parmar, D. Tran, A. Vaswani, P. Koanantakool, P. Hawkins, H. Lee, M. Hong, C. Young, et al., Mesh-tensorflow: Deep learning for supercomputers, Adv. Neural Inf. Process. Syst. 31 (2018).
[50] Z. Jiang, W. Gao, L. Wang, X. Xiong, Y. Zhang, X. Wen, C. Luo, H. Ye, X. Lu, Y. Zhang, et al., HPC AI500: A benchmark suite for HPC AI systems, in: International Symposium on Benchmarking, Measuring and Optimization, Springer, 2018, pp. 10–22.
[51] D. Narayanan, A. Harlap, A. Phanishayee, V. Seshadri, N.R. Devanur, G.R. Ganger, P.B. Gibbons, M. Zaharia, PipeDream: Generalized pipeline parallelism for DNN training, in: Proceedings of the 27th ACM Symposium on Operating Systems Principles, 2019, pp. 1–15.
[52] Z. Jia, M. Zaharia, A. Aiken, Beyond data and model parallelism for deep neural networks, Proc. Mach. Learn. Syst. 1 (2019) 1–13.
[53] data-parallelim, https://en.wikipedia.org/wiki/Data_parallelism.
[54] Z. Ren, Y. Liu, T. Shi, L. Xie, Y. Zhou, J. Zhai, Y. Zhang, Y. Zhang, W. Chen, AIPerf: Automated machine learning as an AI-HPC benchmark, Big Data Min. Anal. 4 (3) (2021) 208–220.
[55] S. Kudo, K. Nitadori, T. Ina, T. Imamura, Prompt report on exa-scale HPL-AI benchmark, in: 2020 IEEE International Conference on Cluster Computing, CLUSTER, IEEE, 2020, pp. 418–419.
[56] B. Zoph, Q.V. Le, Neural architecture search with reinforcement learning, 2016, arXiv preprint arXiv:1611.01578.
[57] Z. Jiang, W. Gao, F. Tang, L. Wang, X. Xiong, C. Luo, C. Lan, H. Li, J. Zhan, HPC AI500 v2. 0: The methodology, tools, and metrics for benchmarking HPC AI systems, in: 2021 IEEE International Conference on Cluster Computing, CLUSTER, IEEE, 2021, pp. 47–58.
[58] L. Breiman, Bagging predictors, Mach. Learn. 24 (2) (1996) 123–140.
[59] Z.-H. Zhou, Ensemble learning, in: Machine Learning, Springer, 2021, pp. 181–210.
[60] T. Ben-Nun, M. Besta, S. Huber, A.N. Ziogas, D. Peter, T. Hoefler, A modular benchmarking infrastructure for high-performance and reproducible deep learning, in: 2019 IEEE International Parallel and Distributed Processing Symposium, IPDPS, IEEE, 2019, pp. 66–77.
[61] S. Farrell, M. Emani, J. Balma, L. Drescher, A. Drozd, A. Fink, G. Fox, D. Kanter, T. Kurth, P. Mattson, et al., MLPerf™ HPC: A holistic benchmark suite for scientific machine learning on HPC systems, in: 2021 IEEE/ACM Workshop on Machine Learning in High Performance Computing Environments, MLHPC, IEEE, 2021, pp. 33–45.
[62] S. Ruder, An overview of gradient descent optimization algorithms, 2016, arXiv preprint arXiv:1609.04747.
[63] R. Farber, AI-HPC is Happening Now, InsideHPC Special Report, InsideHPC, LLC, 2017.
[64] E.A. Huerta, A. Khan, E. Davis, C. Bushell, W.D. Gropp, D.S. Katz, V. Kindratenko, S. Koric, W.T. Kramer, B. McGinty, et al., Convergence of artificial intelligence and high performance computing on NSF-supported cyberinfrastructure, J. Big Data 7 (1) (2020) 1–12.
[65] H. Lee, A. Merzky, L. Tan, M. Titov, M. Turilli, D. Alfe, A. Bhati, A. Brace, A. Clyde, P. Coveney, et al., Scalable HPC & AI infrastructure for COVID-19 therapeutics, in: Proceedings of the Platform for Advanced Scientific Computing Conference, 2021, pp. 1–13.
[66] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: A large-scale hierarchical image database, in: 2009 IEEE Conference on Computer Vision and Pattern Recognition, Ieee, 2009, pp. 248–255.
[67] I. Kandel, M. Castelli, The effect of batch size on the generalizability of the convolutional neural networks on a histopathology dataset, ICT Express 6 (4) (2020) 312–315.
[68] J. Surowiecki, The Wisdom of Crowds, Anchor, 2005.
[69] A. Krizhevsky, One weird trick for parallelizing convolutional neural networks, 2014, arXiv preprint arXiv:1404.5997.
[70] Y. You, J. Li, S. Reddi, J. Hseu, S. Kumar, S. Bhojanapalli, X. Song, J. Demmel, K. Keutzer, C.-J. Hsieh, Large batch optimization for deep learning: Training bert in 76 minutes, 2019, arXiv preprint arXiv:1904.00962.
[71] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, 2014, arXiv preprint arXiv:1412.6980.
[72] J. Duchi, E. Hazan, Y. Singer, Adaptive subgradient methods for online learning and stochastic optimization, J. Mach. Learn. Res. 12 (7) (2011).
[73] Y. Huang, Y. Cheng, A. Bapna, O. Firat, D. Chen, M. Chen, H. Lee, J. Ngiam, Q.V. Le, Y. Wu, et al., Gpipe: Efficient training of giant neural networks using pipeline parallelism, Adv. Neural Inf. Process. Syst. 32 (2019).
[74] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al., TensorFlow: A system for large-scale machine learning, in: 12th USENIX Symposium on Operating Systems Design and Implementation, OSDI 16, 2016, pp. 265–283.
[75] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al., Pytorch: An imperative style, high-performance deep learning library, Adv. Neural Inf. Process. Syst. 32 (2019).
[76] Huawei, Mindspore, https://www.mindspore.cn/.
[77] Nvidia, Nsight system, https://developer.nvidia.com/nsight-systems.
[78] Lawrence Livermore National Laboratory, SLURM, https://slurm.schedmd.com/.
[79] T. Combe, A. Martin, R. Di Pietro, To docker or not to docker: A security perspective, IEEE Cloud Comput. 3 (5) (2016) 54–62.
[80] Nvidia, Nvidia-smi, https://developer.nvidia.com/nvidia-system-management-interface.
[81] iftop, https://en.wikipedia.org/wiki/Iftop.
[82] E. Racah, C. Beckham, T. Maharaj, S. Ebrahimi Kahou, M. Prabhat, C. Pal, Extremeweather: A large-scale climate dataset for semi-supervised detection, localization, and understanding of extreme weather events, Adv. Neural Inf. Process. Syst. 30 (2017).

BenchCouncil Transactions on Benchmarks, Standards and Evaluations 2 (2022) 100084

Contents lists available at ScienceDirect

BenchCouncil Transactions on Benchmarks, Standards and Evaluations

journal homepage: www.keaipublishing.com/en/journals/benchcouncil-transactions-on-benchmarks-standards-and-evaluations/

Research article

CpsMark+: A scenario-oriented benchmark system for office desktop performance evaluation in centralized procurement via simulating user experience

Yue Zhang, Tong Wu *

National Institute of Metrology, China

ARTICLE INFO

Keywords:
Computer benchmarks
Hardware performance evaluation
User experience
Scenario-oriented workloads
Centralized procurement

ABSTRACT

Rapid business expansion of various companies has placed growing demand on office desktops recent decades. However, improper evaluation of system performance and inexplicit awareness of practical use conditions often hamper the efforts to make a consummate selection among multiple alternatives. From the perspective of end users, to optimize the evaluation process of desktop performance in centralized procurement, we present CpsMark+, a coherent benchmark system that evaluates office desktop performance based on simulated user experience. Specifically, CpsMark+ includes scenario-oriented workloads portraying representative user behaviors modeled from the cooperative workflow in modern office routines, and flexibly adapted metrics properly reflecting end-user experience according to different task types. The contrast experiment between state-of-the-art benchmarks demonstrates high sensitivity of CpsMark+ to various hardware components, e.g., CPU, and high repeatability with a Coefficient of Variation less than 3%. In a practical case study, we also demonstrate the effectiveness of CpsMark+ in simulating user experience of tested computer systems under modern office-oriented scenarios for improving the quality of office desktop performance evaluation in centralized procurement.

1. Introduction

Computer performance used to be easily indicated by their hardware configurations. As computer architecture grows more sophisticated, nevertheless, using specifications as a metric will give an incomplete picture of overall computer performance in many practical scenarios [1]. Such an evaluation method is biased and thus cannot catch up with the rapid improvement of computer performance brought by thriving design philosophy. In addition, rapid expansion of computer markets makes it more difficult to identify the system performance.

The above obstacle gives rise to the use of various computer benchmarks. However, most existing benchmarks are unable to meet the performance evaluation requirements in centralized procurement of office computers. Micro and kernel benchmarks are constructed by repeating monotonous operations or running pivotal algorithms from synthetic workloads. These benchmarks merely reflect partial performance of a certain component in a specific system and are primarily utilized by researchers or manufacturers to pursue innovative computer design. While some newer benchmarks, e.g., Business Applications Performance Corporation's SYSmark and Futuremark's PCMark, mainly consist of common business application workloads and are more representative of commercial use, while they fail to offer an overall and scenario-oriented evaluation for general end-user experience [2]. Furthermore, they are not open-source benchmarks, thus the opacity of scoring methodology and workload operations impairs their fairness and transparency, which are essential for centralized procurement.

To address the limitations of SYSmark and PCMark, CpsMark 1.0 [3], an open-source benchmark for microcomputers was developed. However, the design philosophy of CpsMark 1.0 is not user-oriented but emphasizes workload capacity. As a result of such design philosophy, in practice, users complain that workload characterization is biased, and metric measurements are inflexible. In addition, its benchmark methodology is not designed with adequate consideration for office scenarios.

Moreover, it is difficult to precisely grasp the specific needs of end users, let alone individual preferences, especially in centralized procurement. Such inaccessibility makes it unwarranted to formulate the performance evaluation process and limits rational utilization of existing computer benchmarks.

This paper aims to solve the above problems, as well as systematically optimize the process of utilizing benchmarks to evaluate the office desktop performance in centralized procurement. Specifically, we have redeveloped CpsMark+, a novel and coherent benchmark

* Corresponding author.
E-mail addresses: zhyue@nim.ac.cn (Y. Zhang), wut@nim.ac.cn (T. Wu).

https://doi.org/10.1016/j.tbench.2023.100084
Received 8 June 2022; Received in revised form 28 December 2022; Accepted 1 January 2023
Available online 5 January 2023

system that builds a bridge between system performance and simulated user experience in intended usage scenarios, i.e., daily working scenario in modern office. Extensive experiments on multiple real-world tested systems demonstrate high sensitivity and repeatability of CpsMark+ results. Then we used CpsMark+ as a substitute for hardware specifications in quantitatively evaluating the overall computer performance of responsive bids in a real case of centralized procurement. Experimental results show that user experience ratings of the desktops selected by better benchmark score are significantly higher than those selected by the original bid evaluation method, which indicates the effectiveness of CpsMark+ in simulating user experience under modern office-oriented scenario for office desktop performance evaluation in centralized procurement.

The rest of this paper is organized as follows: Section 2 reviews related work and provides our motivation for developing CpsMark+. Section 3 summarizes the challenges in evaluating office computer performance in modern office scenario for centralized procurement. Section 4 describes our methodology and process in developing CpsMark+, as well as extensive experiments for evaluating and comparing CpsMark+ with other related works. Section 5 presents a case study of centralized procurement where we demonstrate the effectiveness of using CpsMark+ as a computer benchmark to simulate user experience in daily office scenario for desktop performance evaluation. Section 6 concludes our work and elicits possible research directions in the future.

2. Background

2.1. Existing benchmarks and metrics

We have reviewed some related works proposed for computer performance evaluation, while most of them have limitations in benchmarking office desktops under modern office scenario for centralized procurement or have not even been designed for commercial use.

SYSmark 2018 [4] adopts real-world third-party software as workloads to evaluate overall computer performance and is widely applied in commercial markets. Usage scenarios are modeled in the form of subjectively grouped job nature like productivity and creativity, which cannot describe cooperation across tasks in a common workflow. In terms of the workloads, most of them are designed to be CPU-intensive and place little pressure on GPU and storage system, making the evaluation insensitive to graphics and I/O performance that might be cared by end users in daily use. Further, system responsiveness and program start-up are isolated and measured by specific applications, thus weakening the realistic reference value of benchmarking results.

PCMark 10 [5] reports an overall score calculated by the geometric mean of tested metrics for the inclusive workloads within each test group. The geometric mean returns a normalized score that treats the performance of each workload equally. This scoring methodology outputs a balanced result of performance evaluation, which neglects the diversity of importance of different workloads and is unable to describe real user experience in a specific scenario.

Phoronix Test System [6] is an open-source and extensible benchmark system that evaluates comprehensive performance of multiple platforms. It includes hundreds of test programs covering a wide range of applications to evaluate various metrics. Nevertheless, the contributors provide little information about benchmarks' logic and internals, especially on how each system is tagged and applied for specific components [7]. Moreover, the benchmark system has numerous functionally overlapping programs for identical system parts and requires complicated dependencies, which makes them too generic and inefficient to be used in centralized procurement.

There are other benchmarks targeting specific application domains. 3DMark [8] mainly describes real-time gaming performance of graphic cards, its dependence of frame rate as the only metric limits further uses in other fields [2]. SPEC CPU 2017 [9] contains a series of floating-point and integer algorithms extracted from the kernel of compute-intensive applications to evaluate the computing performance of CPUs. The workloads are synthetic and biased, making them more suitable for simulative experiments in academic research and industrial development of processors. The Stanford SPLASH benchmark system [10] evaluates parallel algorithms for shared-memory multiprocessors with real scientific workloads, which is of little use for office routines. Micro benchmarks such as STREAM [11] and Imbench [12] solely test single metric like memory bandwidth or latency of individual hardware component through monotonous program operations, which makes them disregard resource allocation and coordination of mixed workload manipulations within the entire computer system [13].

2.2. Our motivation for upgrading CpsMark+

To address the mentioned limitations of SYSmark and PCMark, we released the microcomputer benchmark CpsMark 1.0 in 2014, which evaluates processor performance based on a series of CPU-intensive workloads abstracted from typical computing scenarios [3].

However, the design of CpsMark 1.0 mainly focuses on workload capacity, instead of reflecting end-user experience. The workload operations are designed to be CPU-intensive and isolated from each other, thus it cannot reflect overall performance and user experience in real scenarios, which by contrast, requires workloads to be coherent and interactive. The scoring methodology treats each workload equally and neglects diverse importance of them in practical tasks. In addition, the third-party software used as workloads and the operating system (Windows 7) supported by the benchmark is obsoleted in burgeoning computer-related markets. Generally, these drawbacks merely make CpsMark 1.0 a simple technical reference for an individual customer, while it is powerless to help make purchase decisions according to actual requirements in centralized procurement of office computers.

Over the last few years, the role of benchmarks has been in the spotlight in purchasing computers. Some organization like Bitkom, a Germany's digital association, has proposed the use of benchmarks in tendering of computers [14]. Intel has also recommended some existing benchmarks as the criteria for screening the shortlist from bidders [1]. Inspired by such evolving roles of benchmarks, we redesigned CpsMark 1.0 to a coherent benchmark system by utilizing simulated end-user experience under office-oriented working scenario for better performance evaluation in centralized procurement and finally developed CpsMark+ in 2019.

3. Challenges in evaluating the system performance of office desktops

3.1. Evolution of computer architecture and usage

Researchers and consumers used to compare the performance of diverse computer systems by merely inspecting their hardware specifications. Latency and throughput used to be typical metrics that served us well in computer performance evaluation, since only the size and the content of input data could affect the processing speed of applications at that time [2]. For the sake of performance evaluation, better hardware always led to higher throughput and lower latency so that computer architecture was merely an inorganic combination of individual components.

As computer architecture and usage grow more sophisticated, simple information of computer configurations hardly predicts the program performance in disparate scenarios explicitly [15]. Such transform gradually gives rise to the thriving of numerous benchmarks, which are a system of objective test programs that return the normalized test score compared with the baseline platform by running a series of identical applications or other computer operations. These benchmarks are generally designed to mimic a particular type of workloads on a constant computer system, by which people can be able to compare

the performance of alternative computers under the specific working circumstance.

Nevertheless, modern computer applications increasingly interact with humans, the physical world, and each other—often simultaneously. Some new types of computing tasks like heterogeneous computing [16], for example, can classify different subtasks based on the embedded code segments and automatically assign them to the most suitable computing resources for efficient execution so that the total time consumption of the entire task is minimized. Many tasks operate in parallel and compete for resources internally, which might be a stochastic process and lead to dynamic results. Complicated interactions among tasks, hardware, and humans make it difficult to describe the entire performance of a given system according to a single task or even multiple tasks executed in isolation [2]. Generally, the overall performance of modern computer systems is not solely a function of individual hardware and executed applications, but an intricate integration of hardware architecture, the pattern of software execution and resource allocation, and how humans interact with computer systems [17].

3.2. Obstacles to capturing usage requirements in centralized procurement

Effective evaluation process of computer performance must be built upon an explicit awareness of the intended usage scenario of tested systems, nevertheless, which is especially difficult to obtain for office desktops.

Evaluation of office desktop performance is often massively required in centralized procurement, which is a long and strenuous process where only the opinion of authorities dominates the purchase decision-making. Hence, the decision-making process is usually distant from real stakeholders [18], e.g., the internal customers or the external clients in the case of outsourcing work. The principal of procurement and bidding documents are intensively formulated by management and hardly reflect how procurement items are intended to be used in practice.

Even in the case of individual purchase, compared to traditional electronic products, information of potential usage for modern desktops is still not easy to be directly referenced in the process of performance evaluation, due to the all-round functions and flexible use of modern computers. A game enthusiast who is keen on 3D games, for instance, might also pay attention to computational performance required by a software engineer. Hence, it is hard to capture explicit usage requirements of modern office desktops, which impedes the effective evaluation of system performance, highlighting the importance of how computer benchmarks can precisely reflect end-user experience in specific scenarios.

3.3. Difficulties in reflecting real user experience

In various business domains, a questionnaire is one of the most direct ways to obtain user experience and satisfaction, while like many other similar surveys, it can merely be conducted after the durable use of real end users in practice, which makes it less time-efficient in helping vendors improve their products before releasement or serving as reference when customers are selecting new products. In the field of computers, the rise of various benchmarks solves part of the above problems, however, huge challenges lie in how to precisely reflect user experience without manual intervention.

For a specific computer product, the usage of different potential customer groups could be divergent, which requires an accurate match between benchmark workloads and actual user behaviors. Also, each user may have a different standard in evaluating computer performance, depending on the using habit or product dependency. This phenomenon will influence the perceived user experience without any doubt, and thus requires more considerate design of metrics and scoring methodology. Finally, it is not possible to consummately reflect user experience of computer products with any individual benchmark, because the possible over-specific design will cause the benchmark over-fitting and makes it less applicable for wider use. Therefore, the trade-off between pertinence and universality of benchmarks is also pivotal.

4. The CpsMark+ benchmark tool

In this section, we describe relevant criteria, methodology, and process for developing CpsMark+ in detail. We also carry out analytical and comparative experiments with respect to typical characteristics of computer benchmarks.

4.1. Criteria and design features

Researchers have been theoretically exploring the art of building a consummate benchmark [19,20]. Kistowski et al. [20] assert that all standardized benchmarks are subject to a group of universal criteria, e.g., relevance, repeatability, fairness, and verifiability, which are proved to be necessary. However, in each domain, the criteria are expected to include additional features specific to individual benchmark, depending on its goal, intended usage scenario or other considerations.

The essence of benchmarking office computer performance under daily working scenarios for centralized procurement is to properly evaluate computer systems from a perspective of user experience and describe system performance according to specific purchase demand. In this paper, we propose following benchmark criteria that guide the design of CpsMark+'s features:

- Applications and software manipulations should be scenario-oriented to reflect real user behaviors. Particularly, in centralized procurement, end users can hardly have significant influence on the purchase decision made by authorities, hence the workloads should be closely correlated to behaviors or intended usage that are of interest to end customers in many aspects, e.g., the workload characterization and the input data set.
- Cooperation and diverse importance across tasks should be described. End users usually do not have equal performance requirements for all tasks or even applications involved in an individual task. In practice, if several applications operate towards a common task or purpose, sequence and coherence of them will impact the general working efficiency, since the acceleration of some applications might be more beneficial than that of others.
- Design of metrics should be flexible and account for nonlinearity, which means that composite metrics should not weigh all applications equally. Considering complicated usage of modern desktops, desired metrics of different workloads may vary. In terms of human interaction, for example, a human cannot perceive faster response time beneath some threshold. While for some other tasks, the diversity of execution time on various systems can be ignored.
- The benchmark should be open-source and vendor-neutral. Development of closed-source benchmarks is likely to be manipulated by certain vendors through biased workload design, leading to suspicion [21] and loss of credibility. An open-source benchmark enables public supervision and guarantees the fairness of benchmark results, which is significantly crucial in centralized procurement.

4.2. Entire development process and benchmark framework

Unlike most computer benchmarks, CpsMark+ is designed to be used in centralized procurement, where one single benchmarking result could affect the purchase and use of a specific product for crowds of employees. Hence, during the development process, it is more beneficial to follow an iterative and incremental strategy, instead of top-down principles that formulate schemes at an early stage to make

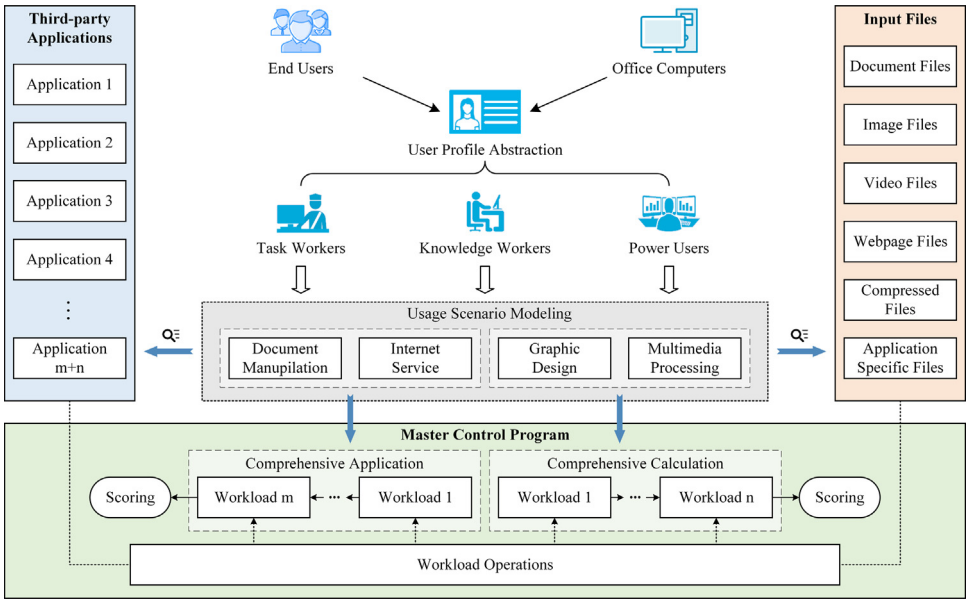


Fig. 1. The main software components and the overall benchmark framework of CpsMark+.

subsequent design right on track. We divide the entire development process into phases, which are associated with relevant checkpoints to guarantee the accomplishment. Within each phase, requirements are elicited from various end users through market research or consultation, then representatives are selected to give feedback on the outcomes of decision-making and implementation. We improve our work based on the feedback and repeat such procedures for each phase. Based on the criteria proposed in Section 4.1, the main software components of CpsMark+ and its overall benchmark framework are depicted in Fig. 1.

CpsMark+ benchmark tool contains three components:

- The automatic setup program, which installs third-party applications and the Master Control Program (MCP) in batches. MCP is responsible for benchmark execution, including test initialization, resource extraction, data integrity check, workload execution, log recording, metric measuring and calculation, and report generation.
- The resource package, including the input files of workload operations.
- The third-party application package, which contains the setups of all third-party applications.

The source code of MCP is maintained online at https://github.com/wanghong3116/CpsMarkPLUS, which is still under further improvement and subject to change. The resource and third-party application packages have been uploaded on the website of National Metrology Data Center of China, which can be accessed online through https://jc.nmdc.ac.cn/view-40-609748.html. Note that CpsMark+ only supports Microsoft Windows 10.

We have not integrated input files, workload applications and the MCP into a unitary package as most commercial benchmarks, which makes our work transparent and easy to be maintained. For the first use of CpsMark+, the trial version of each third-party application is automatically installed on the tested computer system and configured by the execution of an automatic setup program. Likewise, each workload runs independently in the form of complete software, the corresponding application is not merged into the MCP and only receives instructions synchronously from the background of tested computer systems. Such design reduces the influence of the MCP on system performance and enables a clear view of workload conditions provided by logs.

The MCP is devised as a serial layout and contains two separate test modules. Users can initialize the number of iterations to run for eliminating fluctuation of benchmark results. Composed of a sequence of orderly executed workloads, each module independently generates a synthetic score that reflects the performance of inclusive workloads. There is an automatic reboot of the tested computer system between the two modules for eliminating the impacts of varying system status (e.g., cache) on module independence.

4.3. Workloads

CpsMark+ has two independent modules for simulating user experience perceived in modern office scenarios, i.e., Comprehensive Application (CA) and Comprehensive Calculation (CC), which can be optionally selected and run independently during the test. Each of them has a series of workloads executed in a specific order. In this section, we will introduce design and characterization of the workloads within each module in detail.

4.3.1. User profile abstraction of office computers

Chen et al. [22] point out that benchmarks are expected to be associated with real application domains and mirror practical demands in subsistence. Although a large employer may have numerous user segments, appropriate classification could minimize complexity and throw more light on exploring the performance requirement of specific user segment. For the daily usage of desktop computers in modern office scenarios, we abstract the profiles of end users from the perspective of occupation and profession described in Table 1.

Since CpsMark+ has been designed for commercial evaluation of desktop computers used in modern office scenarios, the user profiles summarized in Table 1 exclude those working in laboratories, R&D centers, factories, or telecommuting. In this paper, we mainly focus on most knowledge workers and some part of power users.

4.3.2. Usage scenario modeling and application selection

Employers in a specific department of a company are likely to engage in fixed routine work, thus the performance requirement of a specific task in a homogeneous work section should be more emphasized in centralized procurement of office computers. To highly correlate the design of workloads with the oriented usage scenario of tested computers, we focus on exploring the usage models of intended end users working in daily office scenarios.

According to the abstracted user profiles of office computers, we cluster the usage models into four groups of common office scenarios

Table 1
The profiles of computer end users.

User category	Representative occupations	Performance requirement
Task workers	• Customer service • Front desk consultation • Bank clerks • Data entry specialist • Human resource	• Basic document operations • A single OS-level application • Simple connectivity needs • Static 2D graphics • Few computing occasions
Knowledge workers	• Most students • Teachers and professors • Company administrators • Financial advisors providing multiple advice • Product managers presenting prototypes from multi-angle	• Content creation • Frequent web browsing • Moderately complex application • Moderate scientific computing • Variable multimedia processing like graphics and video • Adequate memory
Power users	• Multimedia designers making high-definition video • Professional architects engaged in complex modeling • Physicians examining delicate 3D medical images	• Complex content creation • Intensive video and 3D graphics processing • Heavy CPU computing • Fast system response • Smooth running of applications

Table 2
The application selection of workloads.

Module	Usage scenario	Application	Version
Comprehensive application	Document manipulation	Microsoft® PowerPoint	2016 (16.0.4266.1003)
		Microsoft® Word	2016 (16.0.4266.1003)
		Microsoft® Excel	2016 (16.0.4266.1003)
		Adobe® Acrobat	DC (19.010.20091)
		WinRAR	5.91 (64-bit)
	Internet service	Google® Chrome	73.0.3683.75
		Microsoft® Outlook	2016 (16.0.4266.1003)
Comprehensive calculation	Graphic design	Autodesk® AutoCAD	2018 (22.0.49.0)
		Adobe® Photoshop	CC 2019 (20.0.1)
	Multimedia processing	Autodesk® 3ds Max	2018 (20.0.0.966)
		Adobe® Premiere	Pro CC 2019 (13.0)
		Adobe® After Effects	CC 2019 (16.0)
		HandBrake	CLI 1.3.0

based on their overall functions within a specific workflow, i.e., document manipulation, Internet service, graphic design, and multimedia processing, which are described as follows:

- The document manipulation scenario contains multiple manipulations towards the documents in common formats, which are involved in most cases of modern business.
- The Internet service scenario mainly includes web browsing and email creation, which are usually auxiliary means in resource acquisition and information communication.
- The graphic design scenario refers to visual expression of ideas and information through the combination of symbols, pictures, and text, which is crucial for product presentation tasks like poster production.
- The multimedia processing scenario relates to utilizing computers for digitizing and integrating graphics, sound, video and other media information in a specific interactive interface, which is widely applied in consulting, marketing and management.

As for workload applications, we select desktop-level office applications based on the metric of popularity. According to the investigation report of office software markets in China by Chinaiern [23], our software market experts select popular and typical applications for each usage scenario in modern office, which are summarized in Table 2.

Since sufficient time is required for workloads to be developed and validated, versions of some applications are not the latest when CpsMark+ was released. In addition, the intended applications of CpsMark+ are the most widely used version instead of the latest one. While some application like WinRAR is up to date because it is feasible to be instantly updated by end users.

4.3.3. Test module construction

While specific selection of usage scenarios ensures high representativeness of the benchmark, grouping applications with similar performance dependencies from various usage scenarios can easily provide an all-sided picture portraying integral performance required by end customers and enhance the usability of the benchmark. Hence, we merge the usage scenarios into two separately running and scored modules as follows:

- Comprehensive Application (CA) module includes the scenarios of document manipulation and Internet service, which reflect light and middleweight use by task or knowledge workers in most business workplaces, where end users might pay more attention to overall performance, response, and smoothness throughout regular use.
- Comprehensive Calculation (CC) module includes the scenarios of graphic design and multimedia processing, which reflect heavyweight use by power users skilled in professional fields, where end users possibly focus on the execution efficiency of CPU-intensive or GPU-intensive computing tasks.

Within each module, in addition to similar performance dependencies, the usage scenarios are highly correlated and tend to appear in a common workflow under daily office scenarios. Further, each usage scenario is given a different weight based on the sum of metrics measured from inclusive workloads. Such approach can ensure a direct and close connection between benchmark results and computer performance required by end users.

4.3.4. Workload components and design details

To reflect the user experience of office computers in modern office, workloads should be not only scenario-oriented but also capable of simulating user behaviors. Therefore, the workload of CpsMark+ is more

than a concept of application automation, but a logical integration of three elements: the input data set extracted from the resource package, the workload operations performed on the input data set through the applications executed by the MCP, and the generated output.

For each workload, the input data set is chosen to functionally reproduce the resources or materials that might be used by end users in modern office scenarios. Specifically, we select raw digital contents or semi-finished project files that are mainly non-structured data such as texts, images, videos, webpages, and other application-specific files, e.g., 3dsMax scene files.

Then we explore basic operating units that frequently appear in the routine use of applications and integrate them into a series of workload operations that can accomplish a common task. We guarantee the completeness of workloads via designing diversified operations that independently generate finished files as output for each application. Moreover, there is no random process in the MCP so that the generated output is uniquely determined by the input data set and the workload operations.

The workload operations of the CA module are briefly described in execution order as follows:

- **Google Chrome.** Simulate users to browse webpages and switch between tabs. Webpages are accessed through locally configured network services. The webpages contain text, pictures, JS (JavaScript) scripts, and flash.
- **Microsoft PowerPoint.** Set the new template style and create slides. Input texts and adjust character formats, alignment, and font size. Add pictures, captions, and typeset. Insert tables and charts with filled data. Browse slides.
- **Microsoft Word.** Input characters, modify titles and character formats, split paragraphs, set the directory, insert pictures, create tables and charts, input data.
- **Microsoft Excel.** Generate and organize data with fixed formula. Classify and enter data under a specific rule. Calculate and sort common statistics. Draw line charts by categories, set titles and styles, adjust size and position. Macro definition and execution.
- **Adobe Acrobat.** Convert PowerPoint, Word, and Excel documents made in previous workloads to PDF files, browse these PDF files page by page.
- **WinRAR.** Compress and decompress mixed files in multiple formats, including images, videos, documents, databases, and log files.
- **Microsoft Outlook.** Simulate users to receive, browse email contents and attachments offline, including Word, Excel, and PowerPoint files. Upload new attachments, edit the body of the email, and reply.

The workload operations of the CC module are briefly described in execution order as follows:

- **Adobe Photoshop.** Use the PSD (Photoshop Document) file to make a vertical poster. Separate target area from the source material and design the layout of layers. In new layers, set titles and captions, add a logo, and adjust its size, coordinates, and transparency. Combine all layers, virtualize the background and merge them into a large picture.
- **Autodesk AutoCAD.** Use the DWG file to draw distributed structure diagrams of buildings. In the main framework, draw structure and vector identification of each area, add coordinates, and mark the size. Change colors of layers and use different line styles. Design wiring, draw pipeline distribution, and flow direction.
- **Autodesk 3ds Max.** Design a 3D model of a whale. Develop the 3D framework, color the texture, add lighting effects, make reflections and shadow effects by calculating light source position, incidence angle, and reflection angle. Produce motion trajectories and movements of the whale model, render segmental frames of action sequences.
- **Adobe Premiere.** Clip and splice source video materials, add lens transition and subtitles, synthesize sound effects, render, and preview the output video.
- **Adobe After Effects.** Add particle explosion effects, render the firework explosion animation sequence of 1800 frames and 30 FPS.
- **HandBrake.** Convert the H.264 encoded source video with 4K resolution to the H.256 encoded target video with 2K resolution, the container format is MP4. Hardware acceleration will be leveraged if enabled.

Within each module, the workloads are executed in the order specified above. The format or even the content of the generated output for some specific applications is identical to that of the input data set for subsequent applications. Such design enables test modules to describe cooperation across tasks throughout a common workflow. For example, the workloads of the CA module simulate the following coherent user behaviors: resource preparation via the Internet, content creation, document processing, and email delivery.

4.4. Metric design and test implementation

Although work efficiency is a pervasive metric in most benchmarks that evaluate computer performance [24] and is widely referenced in helping customers making decisions, unitary metric design may not tell the true story of user experience for the following reasons.

First, people do not have equal performance requirements for all tasks or even for the same portion of an individual task, so that user experience is usually diversified and varying. For instance, professional designers in an advertising agency might pay more attention to the time consumption of multimedia processing, while the user experience of office secretaries is closely related to the response speed and the fluency of frequent document operations.

Second, the perception of user experience is nonlinear and difficult to quantify. In terms of human interaction, humans cannot perceive faster response time beneath a certain threshold, hence further acceleration of the task will not bring better user experience. For example, a frame rate that exceeds the support of a monitor will no longer improve the user experience of a graphics task, while in this case the program execution could be accelerated by a better GPU.

As a result, in the context of CpsMark+, we define work efficiency as the time consumption for systems under test to complete all operations related to user experience within a specific workload, i.e., application launching, input files loading, and basic operating units, which are outlined in Section 4.3.4. Then we take the defined work efficiency as the metric of CpsMark+ and focus on how it can be measured to properly describe the user experience of tested desktops in modern office scenarios.

4.4.1. Method of sampling

To guarantee the pertinence of the metric, CpsMark+ adopts multiple methods to sample the work efficiency of tested computer systems, depending on various workloads. Such a flexible approach can differentiate the user experience by matching the usages of applications with their performance requirements. To be more specific, we predefine runtime as the time spent by each basic operating unit that actively uses system resources, while response time is the time interval between task activation and task completion. The sampling methods are illustrated in Fig. 2.

In terms of the workloads in the usage scenarios of document manipulation (WinRAR excluded) and Internet service, basic operating units are numerous and densely distributed with lightweight resource consumption. Some intervals of them consist of events irrelevant to the evaluation of user experience e.g., temporary retention of screen display, timer interference, which will have an adverse influence on the effectiveness of workloads if they are included in the metric.

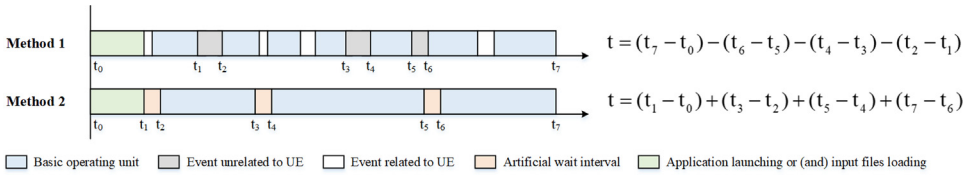


Fig. 2. The two methods of sampling the designed metrics.

However, too many samplings of basic operating units will accumulate the sampling error and cause frequent switches between transient-state and steady-state of program process, which might interfere with the system performance. Hence, we sample the start timestamps and the end timestamps of the entire task and calculate its response time, i.e., $t_7 - t_0$ in Method 1, then we sample the time intervals of irrelevant events and subtract them from the response time as the metric of these workloads.

For the other workloads of CpsMark+, their basic operating units are relatively sparse and have a high concentration of resource consumption. These basic operating units are time-consuming and contribute most of the entire task. In this case, the user experience of end users is more susceptible to the execution speed of a single operation. To accurately measure the runtime, we artificially add extra short waits, e.g., $t_2 - t_1$ in Method 2, between the heavyweight operating units to reset the resource consumption. Finally, we sum the sampled runtime of each basic operating unit as the metric of these workloads.

4.4.2. UI-level vs. API-level automation

Benchmark implementation has a great impact on the test results of the designed metric. There are two primary approaches to automate the execution of workloads, i.e., UI-level and API-level [25,26]. Some benchmarks leverage automated scripts like AutoIt to initiate and navigate applications by simulating mouse clicks or keystrokes [25]. The duration of each task is measured when the completion of the task is detected by application-specific methods. Such an approach mimics practical human interaction at UI level, nevertheless, it instead impedes the accurate reflection of user experience for performance evaluation. Although the estimation of user experience is somewhat subjective, it should be highly relevant to how well computer systems react to or execute the instructions of real end users, however, which might be distorted by a contradictory combination of simulated user behaviors and computer-based metrics.

We choose independent APIs or invoke them from application communication standards, e.g., Component Object Model, to automatically control the execution of each workload. In this case, launching of applications, loading of input files, and basic operating units are implemented through a set of functions, methods, and procedures contained in selected APIs or standards. Compared to the UI-level implementation, our decision to choose API-level implementation provides some tangible benefits as follows:

- **Reduction of irrelevant time measured as metrics.** On the one hand, it takes UI-level implementation a large amount of time to detect the completion of tasks according to the returned signals. For instance, automated scripts may wait for the application to show a pop-up window or may wait for a dialog box to disappear, which requires accurate technical identification. Such a judgment process based on automated scripts is quite time-consuming and significantly falls behind the completion of tasks as perceived by end users. On the other hand, some workload operations themselves take much time for automated scripts to perform. For example, text input might be simulated by continuous keystrokes at a fixed speed, which has identical time consumption on all tested computers. This prolonged simulation accounts for a large proportion of the designed metric and makes the results of measurement diluted by what end users do not value.
- **Less resource consumption and higher test efficiency.** Although some UI-level automated frameworks of benchmarks claim to be lightweight and have little influence on performance, they still consume more computing and memory resources than API-level automation [27]. In addition, API-level automation requires fewer codes to perform and does not need to deal with interface elements. This attribute makes performance evaluation a faster and compact test process and further reduces the overall resource consumption.
- **Greater stability in testing and maintenance.** UI-level automation sometimes gets stuck or goes into endless loops due to UI complexities. For instance, a mouse cursor might miss certain buttons due to the change of resolution, or an unexpected window display may lead to wrong recognition. Some applications are event-driven and can easily enter idle states if there are no users interacting with them [2]. By contrast, API-level automation can guarantee the exact execution of each workload operation and help ease maintenance difficulties brought by external factors [28], e.g., frequent updates of application versions.

4.4.3. Pipeline of metric testing

In CpsMark+, test of the designed metric for a specific workload is performed through the MCP and follows a similar pipeline across all workloads as shown in Fig. 3.

More concretely, for the Nth workload, the MCP first decompresses the resource package and extracts the exclusive input files to a specified location, then an MD5 [29] check is performed towards them to ensure the data integrity. If the MD5 check fails, the test will abort and return to the initialization phase, otherwise, the MCP will move forward to the application execution phase depicted as the dashed rectangle in Fig. 3, where the designed metric T_N is tested. When all the workload operations are finished, an MD5 check is performed towards the generated output. Finally, after a five-second countdown, if there is no user input to interrupt the test, i.e., mouse clicks on the pause button, the MCP will proceed for the next workload until the entire benchmark is completed.

It is worth noting that for the workloads in the usage scenario of document manipulation and Google Chrome, the applications are launched through direct open of the input files, while for the workloads in the usage scenarios of graphic design, multimedia processing, and Microsoft Outlook, the input files are loaded after separate launch of the applications. As a crucial factor affecting the user experience, the speed of application launching is a good indicator of memory and storage performance.

4.5. Scoring methodology

The scoring methodology of benchmarks integrates test results of the designed metric and generates quantified scores that evaluate the overall performance of computer systems. For a commercial benchmark used in centralized procurement, the scoring methodology should provide accurate estimation of the user experience for tested computers to help authorities choose better products from alternatives. For CpsMark+, the design of its scoring methodology meets the following criteria:

- The resulting score does not have significant fluctuation and can remain steady given a constant computer system.

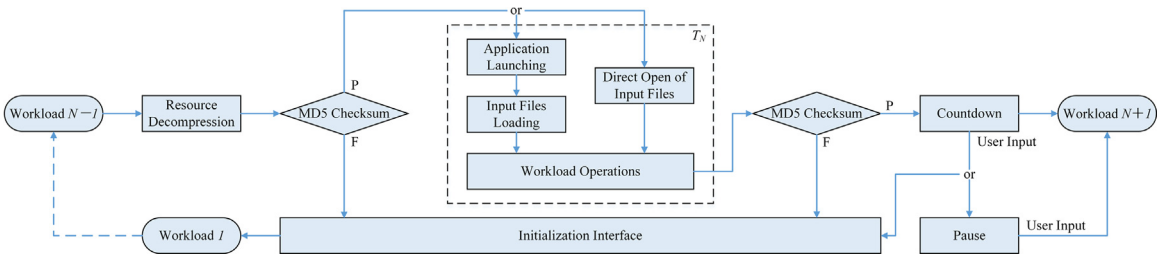


Fig. 3. Intra-workload and inter-workload pipelines of metric testing in CpsMark+.

- The resulting score can sufficiently differentiate the user experience of tested computers with diverse performance.
- The pair-wise relationship between the resulting scores from different computer systems is neutral to the calibration method and the specification of the baseline platform.

Concretely, for each module, we sum the tested metric of each included workload executed on the tested computer system and compare it with the sum of workload metrics tested on the baseline platform. We calculate the ratio value of these two sums and round it to the nearest integer. In this case, a higher score indicates better performance. To be more specific, given the ith module and the number of included workloads N_i, T_j and t_j are the tested metric of the jth workload executed on the tested computer system and the baseline platform, respectively. The resulting score for module i is calculated as follows:

$$S_i = \left[1000 \cdot \frac{\sum_{j=1}^{N_i} t_j}{\sum_{j=1}^{N_i} T_j} \right]$$

Note that we do not take the geometric mean of each score as the overall rating, which places equal weight for each module [30]. Instead, we reserve and separate the score so that end users can flexibly customize the weight of each module when they refer to the benchmark results according to diversified requirements. Within each module, the sum of each tested metric reflects cooperation across workloads and different performance dependencies of them.

4.6. Baseline platform and calibration

As the datum point of the evaluation framework, baseline platforms are prerequisite for most benchmarks. Judicious choice of the baseline platform is of great significance for the resulting score. For instance, an exorbitant configuration of the baseline platform will lead to low sensitivity and weak differentiation of benchmarks, while an inferior one may cause poor repeatability. Hence, at the time of development of CpsMark+, we study the mainstream configurations of office computers purchased in centralized procurement and determine the following configuration for the baseline platform based on performance requirements of the workloads in CpsMark+:

- CPU Model: Intel® Core™ i3-9100 (4 cores, 3.60 GHz, 6 MB L3 cache)
- Graphics: Intel® UHD Graphics 630
- RAM: Kingston® ValueRAM™ 8 GB DDR4 2400 MHz
- Storage: Western Digital® WD Blue™ 1 TB SATA III HDD (6 GB/s, 7200 RPM)
- Chipset: Intel® Z390
- Display Resolution: 1920 × 1080
- OS: Microsoft® Windows® 10

Specifically, to calibrate t_j, we build the baseline platform with brand new parts according to the above hardware configurations and perform a clean installation of the selected operating system. Then we run both modules of CpsMark+ on the baseline platform for 5 independent iterations, the workload-wise calibration t_j is calculated as the median value over the tested metrics of the jth workload from the five runs. Note that since the baseline platform is not a finished product of a computer manufacturer, it is illogical to integrate the tested metrics of all workloads within each module as the module-wise calibration of the baseline platform.

4.7. Benchmark characterization

In this section, we analyze some basic characteristics of CpsMark+ from the perspectives of sensitivity and repeatability, which are two widely used criteria of typical computer benchmarks. Specifically, we have performed extensive test experiments with CpsMark+ on multiple assembled computer systems. Then we analyze the sensitivity of tested module performance to varying hardware characteristics. We also explore the repeatability of workload performance under a constant computer system and stable test environment.

4.7.1. Experimental setup

We alter five different hardware characteristics of a predefined datum point to build the tested computer systems, including the number of CPU cores, CPU frequency, graphics card, storage device, and system memory, which are crucial factors in determining user experience. For each hardware characteristic, we select four configurations with significant pairwise performance differences. They are denoted as Config 1 to Config 4 in ascending order of performance. The detailed configurations of each hardware characteristic are listed in Table 3.

For the configurations of the CPU characteristic, instead of using different processor models, we stick to the CPU model of the datum point and enable different CPU frequencies or numbers of CPU cores by changing BIOS settings. For the configurations of the graphics card, we use the same brand of discrete graphics cards to ensure consistency of graphics drivers and available physical memory. For the configurations of system memory, we all adopt the single-channel mode and only change the memory size of the datum point. The configuration of the datum point is listed as follows:

- CPU Model: Intel® Core™ i7-9700K (8 cores, 3.60 GHz, 12 MB L3 cache)
- Graphics: Nvidia® GeForce® GTX 750
- RAM: Kingston® ValueRAM™ 4 GB DDR4 2666 MHz
- Storage: Seagate® Barracuda® 1TB SATA III HDD (6 GB/s, 5400 RPM)
- Chipset: Intel® Z390
- Display Resolution: 1920 × 1080
- OS: Microsoft® Windows® 10

Notably, for all the experiments in this section, we disable common auxiliary optimization technologies, e.g., Turbo Boost, Hyper-Threading, and Hardware Acceleration, to better highlight the influence of different configurations under various hardware characteristics on benchmark performance from a static perspective. These auxiliary optimization technologies can be enabled in the practical use of CpsMark+.

Table 3
The hardware characteristics and related configurations.

Hardware characteristic	Configuration 1	Configuration 2	Configuration 3	Configuration 4
CPU cores CPU frequency	2-Core 2.0 GHz	4-Core 2.5 GHz	6-Core 3.0 GHz	8-Core 3.5 GHz
Graphics card	Nvidia GeForce GTX 750	Nvidia GeForce GTX 980	Nvidia GeForce GTX 1080	Nvidia GeForce RTX 2080Ti
Storage device	Seagate Barracuda 1TB SATA III 5400RPM HDD	Western Digital Blue 1TB SATA III 7200 RPM HDD	Samsung 860 EVO 250GB SATA III SSD	Samsung 970 PRO 512GB NVMe M.2 SSD
System memory	4 GB	8 GB	16 GB	32 GB

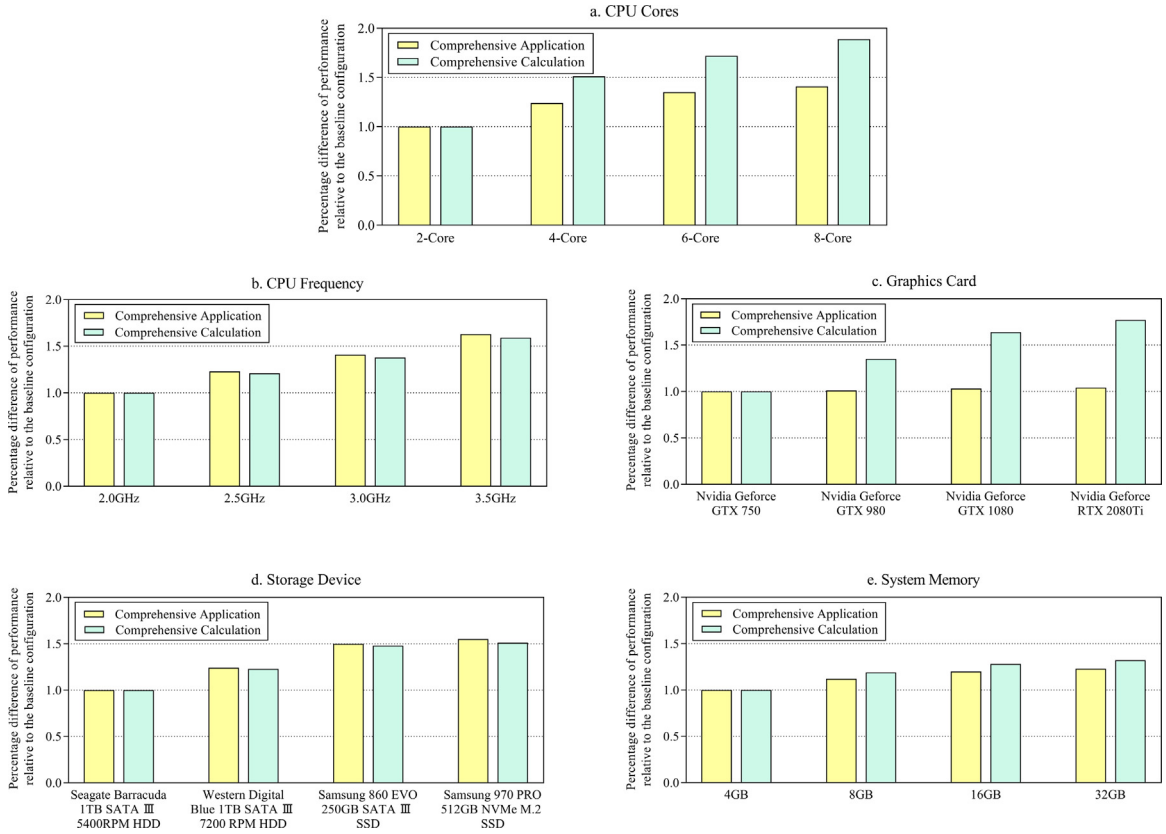


Fig. 4. The sensitivity of the module performance to various hardware characteristics.

4.7.2. Sensitivity analysis

Through evaluating the module performance and the workload performance on tested systems with different levels of configurations, we can explore the sensitivity of CpsMark+ scores to various hardware characteristics. To get strict test results, except for the hardware characteristic under test, the other components of a certain configuration remain identical to the components of the datum point. Specifically, we run CpsMark+ on each configuration for 20 independent iterations with a system reboot and a 15-min interval between each run. In each iteration, we sum the tested metrics of the included workloads for each module, then the average of the sums is adopted as the module performance on a certain configuration. Finally, for each hardware characteristic, we calculate the inverse ratio of the module performance tested on the other three configurations to the module performance tested on the first configuration, i.e., base configuration, respectively. The sensitivity of the module performance and the workload performance of CpsMark+ to various hardware characteristics are shown in Fig. 4 and Table 4, respectively.

Based on the module performance evaluation depicted in Fig. 4, we notice that both modules have a high sensitivity to CPU cores and CPU frequency, the module performance steadily increases as the configurations improve, indicating that both modules can make full use of CPU resources and be significantly affected by more CPU cores and higher CPU frequency. The CC module has a significantly higher sensitivity to the graphics card, the best configuration performs 1.77 times better than the base configuration, while there is no significant difference in the performance of the CA module, which indicates that better graphic cards cannot lead to significant performance improvement of the CA module. Rotation speed and storage media of hard disks also have a great influence on the performance of both modules, since the workloads involve application launching and many I/O operations, while drive interface and protocol contribute less to the module performance. Both modules are relatively less sensitive to system memory than the other hardware characteristics, which indicates that larger size of system memory will bring least significant improvement of both module performance compared to better configurations under other hardware characteristics.

We also notice that the sensitivity of the CC module to most hardware characteristics is higher than the sensitivity of the CA module, since the workloads in the CC module are heavier and have more resource consumption. In addition, as the configurations improve, the growth rate of the module performance slows down, especially for the best configurations, because when configurations exceed some requirement bottleneck of the entire workloads, extra improvement of a single hardware characteristic cannot yield much performance growth.

Table 4
The sensitivity of the workload performance to various hardware characteristics.

CPU cores	Chrome	PowerPoint	Word	Excel	Acrobat	WinRAR	Outlook	Photoshop	AutoCAD	3ds Max	Premiere	After effects	HandBrake
Config 1	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Config 2	1.18	1.21	1.19	1.27	1.23	1.25	1.19	1.48	1.51	1.55	1.49	1.52	1.56
Config 3	1.35	1.37	1.35	1.41	1.36	1.39	1.34	1.73	1.72	1.74	1.69	1.75	1.71
Config 4	1.41	1.43	1.42	1.47	1.44	1.45	1.42	1.89	1.87	1.92	1.91	1.93	1.93
CPU frequency	Chrome	PowerPoint	Word	Excel	Acrobat	WinRAR	Outlook	Photoshop	AutoCAD	3ds Max	Premiere	After effects	HandBrake
Config 1	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Config 2	1.20	1.22	1.21	1.25	1.23	1.26	1.19	1.18	1.19	1.22	1.24	1.23	1.22
Config 3	1.37	1.38	1.39	1.44	1.40	1.43	1.38	1.33	1.35	1.37	1.36	1.34	1.34
Config 4	1.63	1.65	1.64	1.68	1.64	1.67	1.62	1.59	1.57	1.63	1.61	1.58	1.61
Graphics card	Chrome	PowerPoint	Word	Excel	Acrobat	WinRAR	Outlook	Photoshop	AutoCAD	3ds Max	Premiere	After effects	HandBrake
Config 1	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Config 2	1.01	0.99	1.00	1.02	1.01	1.01	0.98	1.36	1.34	1.37	1.35	1.34	1.36
Config 3	1.03	1.00	1.01	1.01	0.99	1.02	1.02	1.66	1.65	1.68	1.66	1.63	1.65
Config 4	1.05	1.04	1.04	1.03	1.02	1.03	1.01	1.77	1.79	1.77	1.75	1.78	1.76
Storage device	Chrome	PowerPoint	Word	Excel	Acrobat	WinRAR	Outlook	Photoshop	AutoCAD	3ds Max	Premiere	After effects	HandBrake
Config 1	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Config 2	1.26	1.25	1.27	1.23	1.24	1.27	1.25	1.25	1.22	1.24	1.25	1.23	1.21
Config 3	1.48	1.47	1.49	1.51	1.50	1.51	1.47	1.46	1.45	1.48	1.49	1.47	1.48
Config 4	1.54	1.55	1.53	1.53	1.55	1.54	1.56	1.51	1.52	1.49	1.50	1.48	1.47
System memory	Chrome	PowerPoint	Word	Excel	Acrobat	WinRAR	Outlook	Photoshop	AutoCAD	3ds Max	Premiere	After effects	HandBrake
Config 1	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Config 2	1.17	1.16	1.17	1.15	1.18	1.14	1.15	1.22	1.19	1.21	1.18	1.22	1.20
Config 3	1.23	1.22	1.25	1.24	1.23	1.22	1.22	1.28	1.27	1.29	1.26	1.25	1.30
Config 4	1.24	1.21	1.26	1.23	1.26	1.23	1.24	1.31	1.29	1.33	1.32	1.28	1.33

As for the sensitivity of the workload performance of CpsMark+ to each hardware characteristic, based on the workload performance evaluation depicted in Table 4, we have observed the same trend as the sensitivity of the module performance. Generally, the performance of all the workloads is highly sensitive to the number of CPU cores, CPU frequency, and the storage devices. The performance of the workloads that require massive GPU-intensive computing, e.g., AutoCAD and Premiere, is more sensitive to graphics cards, compared to the relatively lightweight workloads, e.g., Microsoft Office. However, the performance of some workloads in the CA module, e.g., Excel and WinRAR, is more sensitive to CPU frequency and storage devices, which might be resulted from frequent float point calculations in the RAM and massive document I/O operations in disks triggered by these workloads. We also find out that the performance improvement of most workloads is not significant once the size of system memory reaches 8 GB, which is likely to be the requirement threshold for the workload software to run smoothly.

4.7.3. Repeatability analysis

The repeatability of CpsMark+ is evaluated according to the fluctuation of the module performance and the workload performance tested on the identical computer systems. We leverage Coefficient of Variation (CV), the ratio of the standard deviation to the mean, to indicate the degree of performance fluctuation [31]. To be more specific, for all the experiments under each hardware characteristic, we calculate the CV of the module performance and the workload performance evaluated on the same configuration over 20 independent iterations, respectively. Finally, we aggregate the CV of the module performance under each hardware characteristic and calculate the average CV of the workload performance evaluated on each level of configurations. The results are shown in Fig. 5.

As we can see from the results depicted in Fig. 5, the CV of the module performance under all the hardware characteristics is less than 3%, while the CV of the workload performance under each level of the four configurations is less than 2.5%, which indicates that the overall benchmark results of CpsMark+ are stable and have high consistency under the identical tested computer systems and environment.

Furthermore, for each hardware characteristic, we mark the CV of the module performance under Config 1 and Config 4, respectively. It turns out that except for the results of the CC module under the hardware characteristic of CPU frequency, the best configuration will cause the highest CV of the module performance, while the worst configuration will lead to the lowest CV. Combining the CV of the workload performance with each other, we can conclude that the stability of benchmark results will be improved if the performance of tested configurations exceeds the performance requirements of workloads. As a result, the CV of the module performance under the hardware characteristics of the CPU cores and CPU frequency is relatively high, since CPU frequency of only 2.0 GHz or 2 CPU cores might significantly encumber the performance of tested computer systems.

We also notice that the CV of the heavyweight workload performance is generally higher than the CV of the lightweight workload performance, which is consistent with the previous conclusion, the possible reason is that heavyweight workloads will greatly occupy system resources and lead to unexpected disturbance caused by resource competition between complex program instructions. While Google Chrome is an exception, since the value of its tested metric is relatively small so that it is susceptible to the fluctuation of repetitive experiments. Another finding is that the module performance and the workload performance tested on better configurations will become less volatile, as the performance of high-level configurations might greatly exceed the requirements of workload software. Overall, our benchmark methodology ensures that CpsMark+ is of high sensitivity to provide stable and reliable evaluation results.

4.8. Comparative evaluation against competing benchmarks

In this section, we mainly focus on quantitative and qualitative comparison between CpsMark+ and two commonly used computer benchmarks in commercial field, i.e., SYSmark 2018 and PCMark 10. We explain the experimental and the analytical results in detail, which further highlight the strength and the design philosophy of CpsMark+.

4.8.1. Quantitative comparison

For quantitative comparison, we compare CpsMark+ with SYSmark 2018 and PCMark 10 with respect to the sensitivity and the repeatability of the module performance under various hardware characteristics. We do not select other metrics, e.g., test duration and power consumption, since SYSmark 2018 and PCMark 10 are not open-source

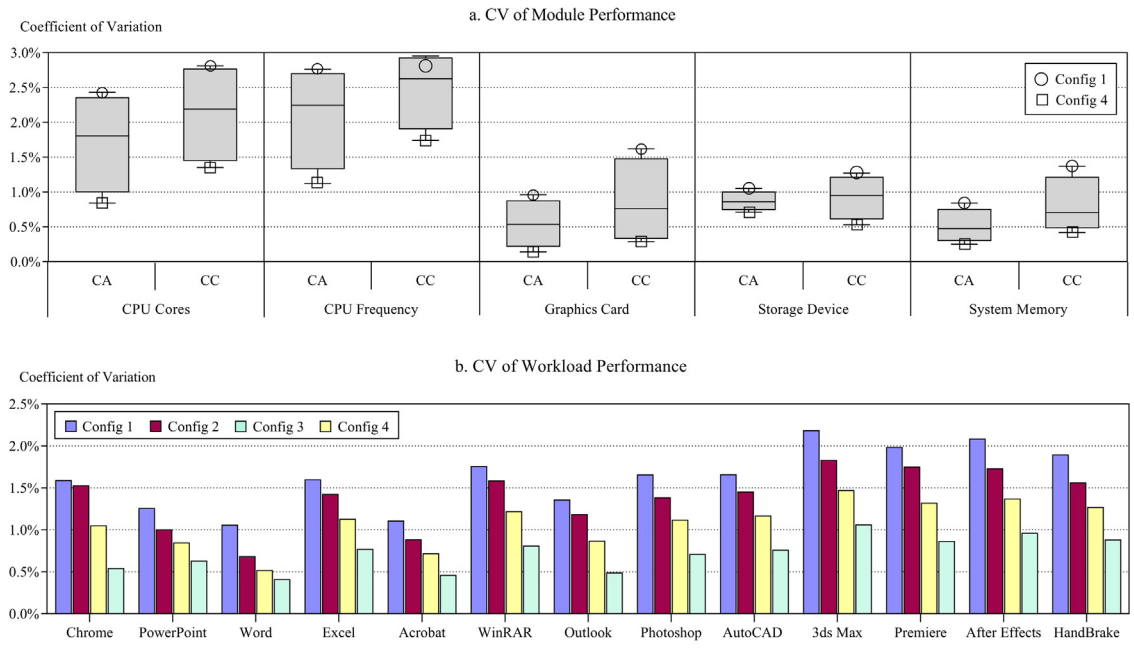


Fig. 5. The repeatability of the module/workload performance under various hardware characteristics.

benchmarks and do not have built-in functions to precisely measure these metrics, which as well makes it impossible to compare the sensitivity and the repeatability of them at a finer granularity, e.g., the level of workload performance. In addition, sensitivity and repeatability are the universal metrics for comparing different benchmarks, even if they possess diverse construction methodologies and usages.

Specifically, we follow the same experimental setup as described in Section 4.7. The modules of SYSmark 2018 include Productivity, Creativity, and Responsiveness, while the modules of PCMark 10 include Essentials, Creativity, and Digital Content Creation. The detailed information about SYSmark 2018 and PCMark 10 is available on their official websites, respectively. Note that in this section, among the three benchmarks, we only compare the sensitivity and the repeatability of the modules that evaluate system performance in similar usage scenarios. Average sensitivity and repeatability (CV in percentage) of the module performance for the three compared benchmarks are summarized in Table 5.

In terms of the sensitivity results depicted in Table 5, among the three modules that evaluate system performance related to document editing and Internet surfing, i.e., the CA module of CpsMark+, the Productivity module of SYSmark 2018, and the Productivity module of PCMark 10, the Productivity module of SYSmark 2018 has the highest sensitivity to all the configurations under each hardware characteristic, since it includes some workloads that have relatively high consumption of system resources, e.g., AutoIT and Shotcut, while the CA module of CpsMark+ has the second highest sensitivity, which is close to the sensitivity of the Productivity module of SYSmark 2018. Among the three modules that evaluate system performance related to multimedia processing and graphics design, i.e., the CC module of CpsMark+, the Creativity module of SYSmark 2018, and the Digital Content Creation module of PCMark 10, the CC module of CpsMark+ is most sensitive to all the hardware characteristics, especially graphics cards, which indicates CpsMark+ can sensitively reflect performance improvement of better GPUs in digital and multimedia processing tasks.

In terms of the repeatability results depicted in Table 5, among the three modules that evaluate system performance related to document editing and Internet surfing, i.e., the CA module of CpsMark+, the Productivity module of SYSmark 2018, and the Productivity module of PCMark 10, the CA module has the highest repeatability, i.e., the lowest CV, across all configurations under each hardware characteristic, while the Productivity module of SYSmark 2018 has the lowest repeatability, i.e., the highest CV. This result is attributed to the UI-level automation of SYSmark 2018, which introduces massive unstable and delayed interactions, e.g., clicking dialog windows. Among the three modules that evaluate system performance related to multimedia processing and graphics design, i.e., the CC module of CpsMark+, the Creativity module of SYSmark 2018, and the Digital Content Creation module of PCMark 10, likewise, the CC module has the highest repeatability, i.e., the lowest CV, across all configurations under each hardware characteristic, which is attributed to the relatively lightweight workloads and the smooth API-level automation of CpsMark+.

Generally, in terms of the modules that evaluate system performance in similar usage scenarios, CpsMark+ exhibits the highest repeatability against state-of-the-art commercial benchmarks, i.e., SYSmark 2018 and PCMark 10, while it also possesses the second highest sensitivity to the hardware characteristics tested in this experiment, which is close to the sensitivity of SYSmark 2018.

4.8.2. *Qualitative comparison*

In this section, we empirically conduct some qualitative comparison of the three benchmarks from the perspectives of workload characterization and scoring methodology.

Firstly, the Responsiveness module of SYSmark 2018 and the Essentials module of PCMark 10 contain a large amount of irrelevant workload operations that cannot precisely simulate user experience perceived in practical usage scenarios of tested computer systems, which is not consistent with the primary attribute of CpsMark+ and accounts for the reason why we exclude them from the above quantitative comparison.

To be more specific, the Responsiveness module of SYSmark 2018 solely measures the response time of program initialization, its workloads consist of a series of sequential application starts and shutdowns, which however, cannot reflect the practical use case in daily office routines and will over amplify the influence of storage devices on the overall performance evaluation based on user experience. On the contrary, each workload of CpsMark+ reflects a common workflow frequently adopted in modern office scenarios and collectively forms typical tasks that are fluent in nature, which exactly justifies our benchmark principal of simulating user experience. Moreover, the Essentials module of PCMark 10 contains the playback of a video with fixed

Table 5
Average sensitivity and repeatability of the module performance for the compared benchmarks.

	CpsMark+ (sensitivity/repeatability)		SYSmark 2018 (sensitivity/repeatability)			PCMark 10 (sensitivity/repeatability)		
CPU cores	CA	CC	Productivity	Creativity	Responsiveness	Essentials	Productivity	Digital content creation
Config 1	1.00/2.43	1.00/2.81	1.00/3.76	1.00/4.97	1.00/4.28	1.00/3.15	1.00/2.78	1.00/4.15
Config 2	1.24/2.15	1.51/2.66	1.28/3.52	1.43/4.35	1.35/3.83	1.19/2.86	1.20/2.62	1.44/3.72
Config 3	1.35/1.46	1.72/1.72	1.41/3.04	1.68/4.06	1.38/3.51	1.30/2.34	1.34/2.17	1.68/3.08
Config 4	1.41/0.84	1.89/1.35	1.47/2.17	1.81/3.87	1.40/3.04	1.33/1.77	1.37/1.56	1.81/2.54
CPU frequency	CA	CC	Productivity	Creativity	Responsiveness	Essentials	Productivity	Digital content creation
Config 1	1.00/2.54	1.00/2.87	1.00/3.88	1.00/5.12	1.00/4.35	1.00/3.25	1.00/2.91	1.00/3.97
Config 2	1.23/2.76	1.21/2.95	1.26/4.02	1.16/5.11	1.13/4.21	1.15/3.11	1.15/2.63	1.13/4.16
Config 3	1.41/1.95	1.38/2.38	1.48/3.34	1.32/4.53	1.19/3.96	1.32/2.48	1.33/2.24	1.32/3.52
Config 4	1.63/1.12	1.59/1.74	1.71/2.73	1.57/4.17	1.22/3.48	1.47/1.93	1.49/1.75	1.49/3.23
Graphics card	CA	CC	Productivity	Creativity	Responsiveness	Essentials	Productivity	Digital content creation
Config 1	1.00/0.96	1.00/1.62	1.00/2.35	1.00/4.16	1.00/3.26	1.00/1.79	1.00/1.48	1.00/3.35
Config 2	1.01/0.64	1.35/1.07	1.04/2.14	1.35/3.25	1.12/2.74	1.02/1.28	1.01/0.81	1.32/2.41
Config 3	1.03/0.43	1.64/0.45	1.05/1.85	1.60/2.64	1.14/2.21	1.03/0.82	1.02/0.59	1.58/1.77
Config 4	1.04/0.14	1.77/0.29	1.05/1.56	1.75/1.83	1.15/1.77	1.03/0.56	1.04/0.37	1.71/1.46
Storage device	CA	CC	Productivity	Creativity	Responsiveness	Essentials	Productivity	Digital content creation
Config 1	1.00/1.05	1.00/1.27	1.00/2.47	1.00/3.47	1.00/2.95	1.00/1.87	1.00/1.52	1.00/2.58
Config 2	1.24/0.88	1.23/0.84	1.29/2.15	1.18/3.12	1.47/2.72	1.24/1.39	1.21/1.07	1.18/2.21
Config 3	1.50/0.84	1.48/1.06	1.53/2.02	1.39/2.85	1.83/2.49	1.39/1.28	1.36/0.89	1.37/1.84
Config 4	1.55/0.71	1.51/0.53	1.65/1.97	1.47/2.34	2.25/2.11	1.47/1.35	1.42/0.81	1.43/1.57
System memory	CA	CC	Productivity	Creativity	Responsiveness	Essentials	Productivity	Digital content creation
Config 1	1.00/0.84	1.00/1.37	1.00/2.20	1.00/3.84	1.00/3.09	1.00/1.85	1.00/1.67	1.00/2.94
Config 2	1.12/0.51	1.19/0.65	1.17/1.75	1.19/2.98	1.23/2.45	1.08/1.26	1.11/1.25	1.14/2.06
Config 3	1.20/0.44	1.28/0.76	1.26/1.58	1.29/3.25	1.25/2.06	1.15/0.99	1.17/0.95	1.25/1.71
Config 4	1.23/0.25	1.32/0.42	1.33/1.33	1.34/2.77	1.26/2.23	1.19/0.63	1.22/0.71	1.30/1.45

duration, thus massive time consumption is included in the calculation of test metrics, which nevertheless, will dilute the contribution of better hardware characteristics to the performance improvement of this module and further reduce the benchmark sensitivity.

Secondly, for each module of PCMark 10, the scoring methodology takes the geometric mean over the test metrics of inclusive workloads, which returns a normalized score that treats the performance of each workload equally and neglects different importance of various workload operations in daily office scenarios. By contrast, as described in Section 4.5, for each module of CpsMark+, the scoring methodology takes the weighted sum over the test metrics of inclusive workloads, which emphasizes the influence of heavy or durable workload performance on simulated user experience and ignores the importance of trivial workload operations that are less involved in the routines of end users.

5. Case study performance evaluation of office desktops using CpsMark+ in a vendor-neutral tendering

In this section, we aim to demonstrate the effectiveness of CpsMark+ in simulating user experience under office-oriented working scenarios for better office desktop performance evaluation in practical centralized procurement. Specifically, in a vendor-neutral tendering of desktop computers for a Chinese company, the tendering was divided into two separate batches with different bid evaluation methods. For the second batch, we combined the original bid evaluation method prepared for the first batch with benchmark scores from CpsMark+ to formulate a new bid evaluation method. The original and the new bid evaluation methods were then independently adopted in the above two tendering batches, respectively. After one-year use of the wining desktops selected by the two bid evaluation methods, we independently investigated the user experience of end users from each tendering batch and collected their ratings. The results show that the desktops purchased in the second batch have significantly higher ratings for user experience, which indicate that the workloads of CpsMark+ can precisely simulate user experience perceived by end users working in modern office-oriented scenarios and enable more targeted performance evaluation for desktops with the above usages.

5.1. Brief introduction of tendering

At the beginning of 2020, a large digital marketing agency in China initialized a centralized procurement to purchase desktop computers for the employees from a functional department and a business department, which are denoted as A and B, respectively. For innovating the traditional tendering policy and validating the effectiveness of CpsMark+, within each department, the procurement was arranged as two separate batches of vendor-neutral tendering with different bid evaluation methods, which are denoted as 1 and 2. The basic information of the four tendering batches are listed in Table 6. Then during the next year, the employees of each department were divided into two groups to use the desktop computers purchased in the two tendering batches, respectively.

Note that in addition to the bid evaluation methods for final decision-making among shortlisted alternatives, we also clarified the minimum technical requirements to preliminarily screen candidates from all bidders, which were based on the standard and high-performance configurations in Bitkom's guideline for IT procurement [14], i.e., Vendor-neutral Tendering of Desktop Computers.

5.2. Improvement of bid evaluation methods

Main difference between the two tendering batches lay in the bid evaluation methods, which were adopted by bid evaluation committee to determine the best bidding product. In this case study, to help authorities purchase desktop computers with better end-user experience at a certain cost and further validate the effectiveness of CpsMark+, we decided to partly replace the straightforward hardware-based scoring rules in the original bid evaluation method with benchmark scores from CpsMark+ to develop the new bid evaluation method.

5.2.1. The original bid evaluation method

The old bid evaluation method consists of 3 sections with a total of 100 points, i.e., the commercial section, the technical section, and the price section. The final score for a certain bid is the sum over the score for each section. Specifically, the score for the commercial section is the direct sum over the score for each included item (0–1 point per

Table 6
Basic information of the four tendering batches.

Tendering batches	Purchase quantity	End users	Primary responsibilities
1A	39	Department A (Functional)	Supportive market research & analysis
2A	39		
1B	46	Department B (Business)	Marketing related service of FMCG
2B	46		

Table 7
The weight of each item within the technical section.

	CPU	Motherboard	Monitor	Memory	Storage	Graphics
1A	15	9	4	11	15	13
1B	20	8	7	11	13	18

item). The score for the technical section is the weighted sum over the score for each included item, which is the weighted average over ratings for various metrics ranked by the importance (0–1 point per metric). Detailed information of the items within each section are listed as follows:

1. Commercial section (3/3 points for 1A/1B)
   - Quality of bid response documents.
   - Efficiency of logistics and query systems.
   - Quality of after-sales service.
2. Technical section (67/77 points for 1A/1B)
   - CPU. Metrics: craftsmanship, number of cores, base frequency, size of L3 cache, Thermal Design Power.
   - Motherboard. Metrics: chipset, expansion slots, structure, BIOS, power supply.
   - Monitor. Metrics: screen size, resolution, brightness, panel type, ports.
   - Memory. Metrics: DDR generations, capacity, operating frequency, CAS latency.
   - Storage. Metrics: HDD/SSD, capacity, rotation speed (for HDD), interface, disk buffer.
   - Graphics. Metrics: integrated/discrete, craftsmanship, architecture, GPU frequency (for discrete graphics).

   The score for the jth item is calculated as follows:

$$Score_j = w_j \cdot \frac{\sum_{i=1}^{n_j}[r_j(i) \cdot \left(1 - \frac{i-1}{n_j}\right)]}{\sum_{i=1}^{n_j}(1 - \frac{i-1}{n_j})}$$

   where n_j is the number of metrics for the jth item, $r_j(i)$ is the rating (0–1 point) for the ith metric of the jth item, w_j is the weight of the jth item predefined by domain experts, which is listed in Table 7.
3. Price section (30/20 points for 1A/1B)
   The lowest quotation among all the bids that meet the minimum technical requirements is defined as the Negotiated Base Price (NBP), then the price section score for a certain bid is the product of the price coefficient and the ratio of the NBP to its quotation. The price coefficients for the tendering batches of 1A and 1B are 30 and 20, respectively.

5.2.2. The new bid evaluation method

In terms of the new bid evaluation method for the tendering batches of 2A and 2B, we introduce benchmark scores from CpsMark+ to replace any items related to system performance in the original technical section, i.e., all the items except for the Monitor item. The weight of the Monitor item and the weights of the commercial and the price sections remain constant. To maintain a total score of 100 points, the benchmark score weights for the tendering batches of 2A and 2B are 63 and 70, respectively.

To calculate the absolute benchmark score from CpsMark+, unlike the item weights within each section in the original bid evaluation method, the weight of the CA/CC module is not predefined by domain experts from the bid evaluation committee, instead it is assigned as the average value of the survey results from the real end users of both departments. The weights of the CA/CC module for the tendering batches of 2A and 2B turn out to be 0.71/0.29 and 0.12/0.88, respectively. Then the absolute benchmark score from CpsMark+ for each tendering batch is defined as follows:

$$Score_{cps} = \sqrt[\Sigma_{i=1}^{2} w_i]{\prod_{i=1}^{2} s_i^{w_i}}$$

where w_i is the weight of the ith module, s_i is the median score of the ith module over 5 independent tests on a certain bidding product.

To scale the absolute benchmark score from CpsMark+ for better reflection of relative performance among various bidding products, we adopted a similar strategy as in the price section. Specifically, the best absolute benchmark score among all the bids that meet the minimum technical requirements is defined as the Negotiated Maximum Performance (NMP), then the final benchmark score for a certain bid is the product of the benchmark score weight and the ratio of its absolute benchmark score to the NMP. Finally, the score for the new technical section is the direct sum over the final benchmark score and the score for the Monitor item.

5.3. Effects of introducing benchmark scores from CpsMark+

To evaluate the effects of introducing benchmark scores from CpsMark+ as part of the new bid evaluation method, for the winning bids purchased in the tendering batches of 1A/1B and 2A/2B, we performed a comparative analysis towards the one-year user experience rated by the respective end users.

5.3.1. Evaluation protocols of user experience

We first formulated the explicit evaluation protocols for rating user experience of office desktops in modern office scenarios. The ISO 9241 standard [32] of human–computer interaction defines usability as "the extent to which a product can be used by specific users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use". We defined user experience of the winning bids in a similar way as the usability defined in the ISO 9241 standard. Since for all the bids that meet the minimum technical requirements, the effectiveness of the products in fulfilling the tasks specified by the tenders is guaranteed, we mainly focused on the following two metrics:

(1) Efficiency, i.e., the user-perceived time consumption for software and applications to achieve specified goals. The rating is scaled as "very efficient" (5 points), "somewhat efficient" (4 points), "neutral" (3 points), "somewhat inefficient" (2 points), or "very inefficient" (1 point).

(2) Smoothness, i.e., the user-perceived overall smoothness in daily use of software or applications, including jank, launching speed, delay, and response to instructions. The rating is scaled as "very smooth" (5 points), "somewhat smooth" (4 points), "neutral" (3 points), "somewhat unsmooth" (2 points), or "very unsmooth" (1 point).

In terms of the rating items, we surveyed each department to find out the software or applications frequently used by most end users

Table 8
The rating items and corresponding weights.

	MySQL	Excel	Power BI	Photoshop	Premiere	After effects	Internet explorer	Word	PowerPoint	Lark [33]
1A/2A	0.18	0.19	0.13	0	0	0	0.09	0.16	0.12	0.13
1B/2B	0	0	0	0.22	0.18	0.25	0.14	0.10	0.05	0.06

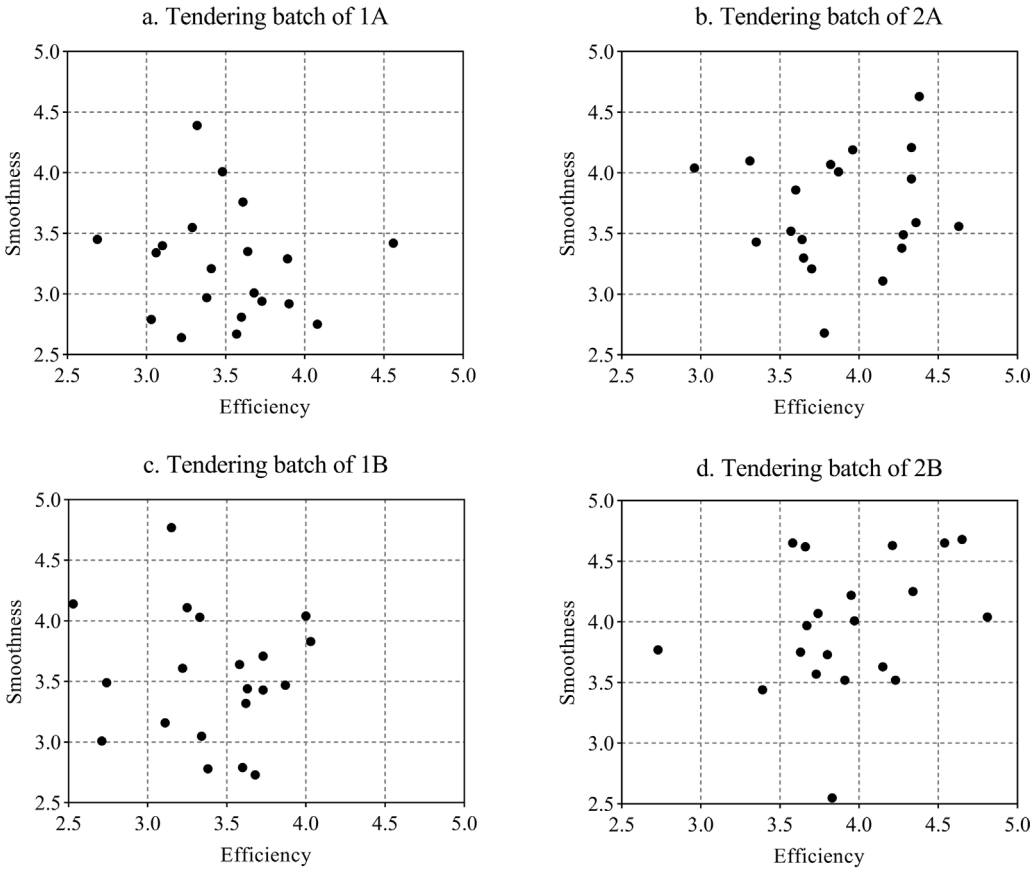


Fig. 6. The distributions of user experience ratings for the winning bids.

within one year after the procurement. Then we gave them different weights according to the average hours of use over the entire department, which are listed in Table 8.

For each tendering batch, i.e., 1A, 2A, 1B, and 2B, we randomly invited 20 end users from the corresponding group of their department to independently rate the user experience of the desktop computers purchased in this tendering batch. The questionnaires adopted for rating the user experience are similar as CSAT [34]. For each desktop computer, the total score for each metric of the user experience is the weighted sum over the metric ratings for all the items.

5.3.2. Evaluation results

The distributions of user experience ratings for the winning bids from the four tendering batches are shown in Fig. 6. As we can see from the results, for both metrics of the user experience, the ratings from all surveyed end users are between 2.5 and 5 points. Specifically, the ratings for both user experience metrics of the winning bids from the tendering batches of 1A/1B are mostly between 2.5 and 4 points, while the ratings from the tendering batches of 2A/2B are mostly between 3 and 4.5 points, which indicates that the user experience of the desktop computers selected by the new bid evaluation method is improved to some extent.

Table 9 shows some descriptive statistics of the above user experience ratings and the average quotation for the desktops purchased from each tendering batches. For the tendering batches of 1A/2A, the efficiency and the smoothness ratings for the winning bids are 3.51/3.90 points and 3.23/3.69 points, with an increase of 11.11% and 14.24%, respectively. For the tendering batches of 1B/2B, the efficiency and the smoothness ratings for the winning bids are 3.40/3.93 points and 3.53/3.96 points, with an increase of 15.59% and 12.18%, respectively.

Although the rating results of user experience demonstrate the effectiveness of CpsMark+ in identifying office desktops with better user experience under modern office-oriented scenarios, the analysis so far has only told part of the story for evaluating the effects of introducing benchmark scores from CpsMark+ in centralized procurement, since pricier bids generally tend to deliver better system performance, which will cause a much higher budget. To this end, we also consider the average quotation for the winning bids from each tendering batch, which is 5316/5562 CNY and 6465/6948 CNY for the tendering batches of 1A/2A and 1B/2B, with an increase of 4.63% and 7.47%, respectively. Note that in this paper, charges for other services, e.g., logistics and insurance, are excluded from the average quotation. Apparently, the higher average quotation of the winning bids leads to more significant increase of user experience ratings. This result demonstrates that the new bid evaluation method based on benchmark scores from CpsMark+ can help authorities select the bid with better user experience and higher cost-effectiveness in the centralized procurement of office desktops.

5.3.3. Statistical analysis

In this case study, we randomly selected 20 end users from each tendering batch for higher survey efficiency and minimizing the rating

Table 9
Descriptive statistics of the user experience ratings and the average quotation for the winning bids.

		1A	2A	1B	2B
Efficiency	Mean (%)	3.51 (70%)	3.90 (78%)	3.40 (68%)	3.93 (79%)
	95% confidence interval	[3.32–3.71]	[3.69–4.10]	[3.18–3.61]	[3.70–4.15]
Smoothness	Mean (%)	3.23 (65%)	3.69 (74%)	3.53 (71%)	3.96 (79%)
	95% confidence interval	[3.02–3.45]	[3.47–3.91]	[3.28–3.78]	[3.71–4.22]
Average quotation per computer, CNY		5316	5562	6465	6948

Table 10
Results of the *p*-value in significance tests (at a 5% significance level).

		1A	2A	1B	2B
Efficiency	Normality	0.8978	0.5410	0.1805	0.5569
	Homogeneity of variance	0.8486		0.8741	
	Student's t-test	0.0070		0.0001	
Smoothness	Normality	0.1643	0.8280	0.6373	0.0643
	Homogeneity of variance	0.9769		0.8455	
	Student's t-test	0.0035		0.0165	

deviation due to the subjective evaluation of user experience. Hence, we perform further statistical analysis to explore potential significant changes of user experience ratings within the whole populations from the tendering batches of 2A/2B. The results of significance tests are shown in Table 10.

According to the Shapiro–Wilk test, the normality for all the distributions of user experience ratings is accepted, which indicates that user experience of the winning bids from each tendering batch is concentrated within a certain range. Then we conduct a two-tailed F test to infer the homogeneity of variance between user experience ratings from 1A and 2A, as well as 1B and 2B. Specifically, all the results accept the null hypothesis, which is possibly attributed to the similar responsibilities of employees from the same department.

The results of the student's t-test also infer a significant change of user experience ratings within the whole populations from the tendering batch of 2B. Specifically, the p-value of the student's t-test for efficiency ratings from the tendering batches of 1B/2B is just 0.0001, which suggests that a significant change of user experience perceived by all the employees from the tendering batch of 2B exists with a large probability. The possible reason is that system performance of the winning bids from the tendering batch of 2B breaks through requirement bottleneck of the routine tasks in department B.

5.3.4. User experience of items excluded from CpsMark+

Although we have seen significant improvements in user experience of the winning bids selected by the new bid evaluation method, the rating items for user experience evaluation partly overlap with the workloads of CpsMark+. Without loss of generality, we conduct a comparative analysis to further validate the effectiveness of CpsMark+ in simulating user experience of tested computer systems with respect to software or applications that are not included in its workloads.

Specifically, for each rating item that is not adopted as the workload application of CpsMark+, we collect and average its user experience metrics over the winning bids selected by the original and the new bid evaluation methods, respectively. The results are shown in Fig. 7 and Table 11.

According to the above results, under the workloads that are not included in CpsMark+, user experience of the office desktops selected by the new bid evaluation method also improves by varying degrees. For example, in terms of heavy workloads, the average ratings for efficiency and smoothness of MySQL increase by 22.95% and 26.56%, respectively. The similar trend of user experience improvement is also observed with respect to more lightweight workloads, e.g., Power BI, Internet Explorer, and Lark. These results suggest that the workloads of CpsMark+ are sufficiently representative for simulating user experience of tested computer systems perceived under a wide range of workloads.

6. Conclusions

This paper presents CpsMark+, a scenario-oriented benchmark system that quantitively evaluates the overall performance of office desktops in centralized procurement. Considering the proposed challenges in benchmarking desktops under practical usage scenarios for centralized procurement, the workloads of CpsMark+ are designed to be scenario-oriented and can simulate user experience of tested computer systems perceived by end users working in modern-office scenarios. The metrics testing and the scoring methodology are flexibly adjusted based on each individual workload. Extensive experiments on multiple real-world tested computer systems demonstrate high sensitivity and repeatability of benchmark scores from CpsMark+, compared to SYSmark 2018 and PCMark 10. From the perspective of end users,

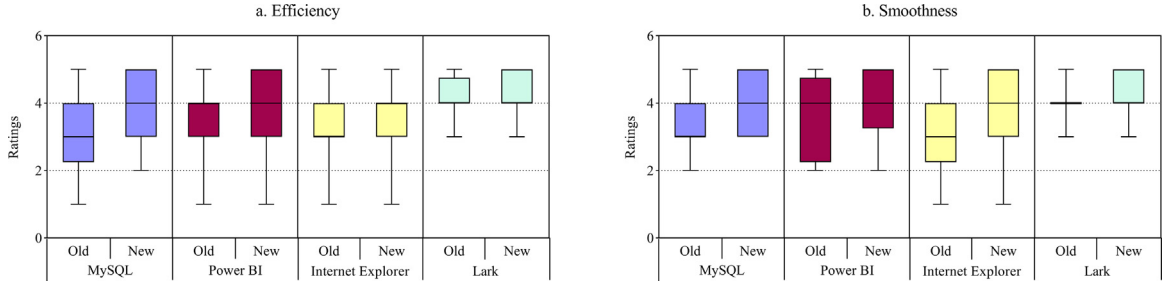


Fig. 7. User experience ratings for software or applications absent in CpsMark+.

Table 11
Average user experience ratings for software or applications absent in CpsMark+.

		MySQL	Power BI	Internet Explorer	Lark
Efficiency	Old bid evaluation method	3.05	3.50	3.43	4.20
	New bid evaluation method	3.75	3.80	3.60	4.28
Smoothness	Old bid evaluation method	3.20	3.55	3.20	3.98
	New bid evaluation method	4.05	3.95	3.55	4.30

in a practical centralized procurement of office desktops, by replacing the original bid evaluation method with benchmark scores from CpsMark+ and comparing user experience ratings for the winning bids selected by the two bid evaluation methods, we also demonstrate the effectiveness of using CpsMark+ to simulate user experience of tested systems in modern-office scenarios for better evaluation of office desktop performance in centralized procurement.

Our work provides a general idea to design computer benchmarks used in other usage scenarios and helps further explore the benefits of introducing benchmark scores in traditional bid evaluation methods for centralized procurement of office desktops. In the future, we will focus on designing parallel workloads that contain more complex interactions and involving other metrics, e.g., battery life or energy efficiency.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the National Key R&D Program of China under grant number 2018YFF0212106. We are thankful to the purchasing manager in the centralized procurement and all the employees that participated in the user experience survey.

References

[1] U. Norf, The role of benchmarks in the public procurement of computers, 2019, https://www.intel.co.uk/content/dam/www/public/us/en/documents/white-papers/role-of-benchmarks-white-paper.pdf.
[2] S.M. Pieper, J.M. Paul, M.J. Schulte, A new era of performance evaluation, Computer 40 (9) (2007) 23–30.
[3] Release of CpsMark 1.0, 2014, https://read01.com/4aAj54.html#.YWL7O9pBxPa.
[4] SYSmark 2018, 2018, https://bapco.com/products/sysmark-2018/.
[5] PCMark 10, 2017, https://benchmarks.ul.com/pcmark10.
[6] Phoronix test system, 2022, https://www.phoronix-test-system.com/.
[7] A. Martin, V. Marangozova-Martin, Automatic benchmark profiling through advanced trace analysis, in: European Conference on Parallel Processing, Springer, Cham, 2016, pp. 63–74.
[8] 3Dmark, 2022, https://benchmarks.ul.com/3dmark.
[9] J. Bucek, K.D. Lange, J.v. Kistowski, SPEC CPU2017: Next-generation compute benchmark, in: Companion of the 2018 ACM/SPEC International Conference on Performance Engineering, 2018, pp. 41–42.
[10] S.C. Woo, M. Ohara, E. Torrie, J.P. Singh, A. Gupta, The SPLASH-2 programs: Characterization and methodological considerations, ACM SIGARCH Comput. Archit. News 23 (2) (1995) 24–36.
[11] J.D. McCalpin, Stream benchmark, 1995, Link: www.cs.virginia.edu/stream/ref.html#what, 22(7).
[12] L.W. McVoy, C. Staelin, Lmbench: Portable tools for performance analysis, in: USENIX Annual Technical Conference, 1996, pp. 279–294.
[13] G. Lu, X. Lin, R. Zhou, Mbench: Benchmarking a multicore operating system using mixed workloads, in: BPOE, Springer, Cham, 2015, pp. 50–63.
[14] F. Felicia, Vendor-neutral tendering of desktop computers, 2019, https://www.itk-beschaffung.de/sites/beschaffung/files/2021-03/200128_lf_vendor-neutral-tendering-of-desktop-computers_en.pdf.
[15] A. Tarvo, S.P. Reiss, Using computer simulation to predict the performance of multithreaded programs, in: Proceedings of the 3rd ACM/SPEC International Conference on Performance Engineering, 2012, pp. 217–228.
[16] S. Mittal, J.S. Vetter, A survey of CPU–GPU heterogeneous computing techniques, ACM Comput. Surv. 47 (4) (2015) 1–35.
[17] Y. Wang, V. Lee, G.Y. Wei, D. Brooks, Predicting new workload or CPU performance by analyzing public datasets, ACM Trans. Archit. Code Optim. (TACO) 15 (4) (2019) 1–21.
[18] S. Vagstad, Centralized vs. decentralized procurement: Does dispersed information call for decentralized decision-making? Int. J. Ind. Organ. 18 (6) (2000) 949–963.
[19] K. Huppler, The art of building a good benchmark, in: Technology Conference on Performance Evaluation and Benchmarking, Springer, Berlin, Heidelberg, 2009, pp. 18–30.
[20] J. v. Kistowski, J.A. Arnold, K. Huppler, K.D. Lange, J.L. Henning, P. Cao, How to build a benchmark, in: Proceedings of the 6th ACM/SPEC International Conference on Performance Engineering, 2015, pp. 333–336.
[21] U. Gordon, PCWorld, in: AMD Accuses BAPCo and Intel of Cheating with Sysmark Benchmarks, 2016, https://www.pcworld.com/article/419213/amd-accuses-bapco-and-intel-of-cheating-with-sysmark-benchmarks.html.
[22] Y. Chen, F. Raab, R. Katz, From tpc-c to big data benchmarks: A functional workload model, in: Specifying Big Data Benchmarks, Springer, Berlin, Heidelberg, 2012, pp. 28–43.
[23] In-depth research of China office software market, 2021, http://www.chinaiern.com/baogao/scbg/2953763.shtml?bd_vid=6645286086633733527.
[24] A. Crolotte, Issues in benchmark metric selection, in: Technology Conference on Performance Evaluation and Benchmarking, Springer, Berlin, Heidelberg, 2009, pp. 146–152.
[25] T. Nguyen, P. Calyam, R.B. Antequera, Benchmarking in virtual desktops for end-to-end performance traceability, in: 2015 IFIP/IEEE International Symposium on Integrated Network Management, IM, IEEE, 2015, pp. 1268–1273.
[26] S. Taheri, L.A. Beni, A.V. Veidenbaum, A. Nicolau, R. Cammarota, J. Qiu…, M.R. Haghighat, WebRTCbench: a benchmark for performance assessment of webRTC implementations, in: 2015 13th IEEE Symposium on Embedded Systems for Real-Time Multimedia (ESTIMedia), IEEE, 2015, pp. 1–7.
[27] B. Daniel, Q. Luo, M. Mirzaaghaei, D. Dig, D. Marinov, M. Pezzè, Automated GUI refactoring and test script repair, in: Proceedings of the First International Workshop on End-To-End Test Script Engineering, 2011, pp. 38–41.
[28] T.E. Vos, P.M. Kruse, N. Condori-Fernández, S. Bauersfeld, J. Wegener, Testar: Tool support for test automation at the user interface level, Int. J. Inf. Syst. Model. Des. (IJISMD) 6 (3) (2015) 46–83.
[29] R. Rivest, The MD5 message-digest algorithm (No. rfc1321), 1992.
[30] P.J. Fleming, J.J. Wallace, How not to lie with statistics: the correct way to summarize benchmark results, Commun. ACM 29 (3) (1986) 218–221.
[31] A.G. Bedeian, K.W. Mossholder, On the use of the coefficient of variation as a measure of diversity, Organ. Res. Methods 3 (3) (2000) 285–297.
[32] T. Jokela, N. Iivari, J. Matero, M. Karukka, The standard of user-centered design and the standard definition of usability: analyzing ISO 13407 against ISO 9241-11, in: Proceedings of the Latin American Conference on Human–Computer Interaction, 2003, pp. 53–60.
[33] Lark, 2022, https://www.larksuite.com.
[34] V. Mittal, C. Frennea, Customer Satisfaction: A Strategic Review and Guidelines for Managers, in: MSI Fast Forward Series, Marketing Science Institute, Cambridge, MA, 2010.

Yue Zhang born in 1995, research assistant. His main research includes IT benchmarks, AIops. Now he is a Ph.D. candidate in Renmin University of China.

Tong Wu born in 1975, associate research fellow. His main research includes IT benchmarks, computer architecture and EMC. Now he works in National Institute of Metrology, China.

BenchCouncil Transactions on Benchmarks, Standards and Evaluations 2 (2022) 100087

Contents lists available at ScienceDirect

BenchCouncil Transactions on Benchmarks, Standards and Evaluations

journal homepage: www.keaipublishing.com/en/journals/benchcouncil-transactions-on-benchmarks-standards-and-evaluations/

Research Article

Optimizing the sparse approximate inverse preconditioning algorithm on GPU☆

Xinyue Chu, Yizhou Wang, Qi Chen, Jiaquan Gao *

Jiangsu Key Laboratory for NSLSCS, School of Computer and Electronic Information, Nanjing Normal University, Nanjing 210023, China

ARTICLE INFO

Keywords:
Sparse approximate inverse
Preconditioning
CUDA
GPU

ABSTRACT

In this study, we present an optimization sparse approximate inverse (SPAI) preconditioning algorithm on GPU, called GSPAI-Opt. In GSPAI-Opt, it fuses the advantages of two popular SPAI preconditioning algorithms, and has the following novelties: (1) an optimization strategy is proposed to choose whether to use the constant or non-constant thread group for any sparse pattern of the preprocessor, and (2) a parallel framework of optimizing the SPAI preconditioner is proposed on GPU, and (3) for each component of the preconditioner, a decision tree is established to choose the optimal kernel of computing it. Experimental results validate the effectiveness of GSPAI-Opt.

1. Introduction

Given their many-core structures, graphic processing units (GPUs) have become an important resource for scientific computing in recent years. Following the introduction of the programming interfaces such as the compute unified device architecture (CUDA) by NVIDIA in 2007 [1], GPUs have been increasingly used as tools for high-performance computation in many fields [2–8].

Sparse approximate inverse (SPAI) preconditioners based on the Frobenius norm minimization have proven to be effective in improving the convergence of iterative methods based on Krylov subspaces, e.g., the generalized minimal residual method (GMRES) [9] and the biconjugate gradient stabilized method (BiCGSTAB) [10]. However, due to the high cost of constructing the SPAI preconditioners, many researchers have attempted to accelerate the SPAI preconditioner construction on GPU. Gao et al. follow Chow's work [11], and use a sparse approximate inverse of A as the preconditioner in [12]. Rupp et al. [13] show several static and dynamic SPAI implementations on GPU. In [14], Dehnavi et al. propose a static SPAI preconditioner on GPU called GSAI. Recently, He and Gao et al. [15] propose a GPU-based static SPAI preconditioning algorithm called SPAI-Adaptive, and verify the effectiveness of SPAI-Adaptive for large-scale matrices. However, when the number of nonzero entries in each column of the preconditioner has significant difference, the performance of SPAI-Adaptive is greatly decreased. Furthermore, He and Gao et al. [16] present a sorted static SPAI preconditioning algorithm, called GSPAI-Adaptive, in order to avoid the drawback of SPAI-Adaptive.

SPAI-Adaptive and GSPAI-Adaptive both can be applied to large-scale matrices, and have their own advantages. When the difference in the nonzero number of each column of the preconditioner is small, the performance of SPAI-Adaptive is generally better than that of GSPAI-Adaptive; when the nonzero number of each column of the preconditioner has significant difference, SPAI-Adaptive has worse performance than GSPAI-Adaptive. For example, assuming that $n2_k$ is the nonzero number of the kth column of the preconditioner, $n2max = \max_k\{n2_k\}$, and $n2avg = \sum_{k=1}^{n} n2_k/n$, where n is the row number of the preconditioner, we take two integers α and β, which satisfy $2^{\alpha-1} < n2max \leqslant 2^{\alpha}$ and $2^{\beta-1} < n2max \leqslant 2^{\beta}$, respectively. If $\alpha = \beta$, we say that the difference in the nonzero number of each column of the preconditioner is small; if $\alpha - \beta \geqslant 3$, we say that the nonzero number of each column of the preconditioner has significant difference. However, when the difference is large but not significant, which one of SPAI-Adaptive and GSPAI-Adaptive has better performance? For example, $1 \leqslant \alpha - \beta < 3$. There are no conclusions in [15,16].

Inspired by these observations, we further investigate how to highly optimize the static SPAI on GPU in this paper. Utilizing the advantages of SPAI-Adaptive and GSPAI-Adaptive, we propose an optimized SPAI preconditioning algorithm on GPU, called GSPAI-Opt. Compared to SPAI-Adaptive and GSPAI-Adaptive, the proposed algorithm has the following distinct characteristics:

- First, an optimization strategy is presented. Using this strategy, for a given sparsity pattern of the preconditioner, we can obtain the optimization scheme of choosing whether to use the constant or nonconstant thread-group size to calculate the preconditioner.
- Second, when the constant thread-group size is applied, for each one of main components of the preconditioner such as finding indices I and J, constructing the local submatrix, decomposing the

☆ The research has been supported by the Natural Science Foundation of China under grant number 61872422.
* Corresponding author.
E-mail addresses: 2316607219@qq.com (X. Chu), 1966224230@qq.com (Y. Wang), 1337223917@qq.com (Q. Chen), springf12@163.com (J. Gao).

https://doi.org/10.1016/j.tbench.2023.100087
Received 11 October 2022; Received in revised form 26 February 2023; Accepted 26 February 2023
Available online 3 March 2023

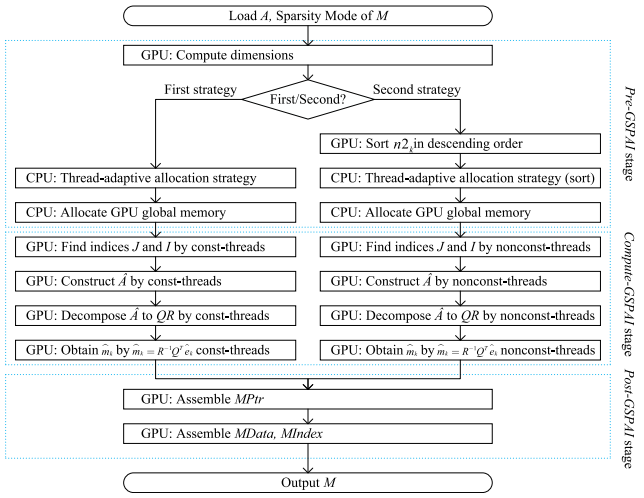


Fig. 1. Parallel framework of GSPAI-Opt.

local submatrix into QR, and solving the upper triangular linear system, a decision tree is established to choose the optimization kernel of calculating it.

- Third, when using the nonconstant thread-group size, for each one of some components of the preconditioner such as decomposing the local submatrix into QR and solving the upper triangular linear system, a decision tree is constructed to choose the optimization kernel to calculate it.
- Finally, GSPAI-Opt can apply to any sparsity pattern of the preconditioner, not just the same sparsity pattern as A.

The experimental results show that GSPAI-Opt is effective, and efficiently fuses the advantages of SPAI-Adaptive and GSPAI-Adaptive, and outperforms the static SPAI preconditioning algorithm in the ViennaCL library [13], the recent SPAI-Adaptive [15] and GSPAI-Adaptive [16].

2. Optimizing SPAI on GPU

We present an optimization sparse approximate inverse preconditioning algorithm on GPU, called GSPAI-Opt. Fig. 1 lists the parallel framework of GSPAI-Opt, which is composed of the following stages.

- *Pre-GSPAI* stage: Compute the dimensions, choose whether to allocate the constant thread-group size or nonconstant thread-group size for each column of the preconditioner according to the proposed optimization strategy, and allocate the global memory of GPU;
- *Compute-GSPAI* stage: Find indices J_k and I_k, construct local submatrix $\widehat{A}_k$, decompose $\widehat{A}_k$ into $Q_k R_k$, and solve $R_k \widehat{m}_k = Q_k^T \widehat{e}_k$;
- *Post-GSPAI* stage: Assemble the preconditioner M in the compressed sparse column (CSC) storage format.

Based on the sparsity pattern of the preconditioner, when the thread allocation strategy with the constant thread-group size is more suitable for computing the preconditioner, the thread-adaptive allocation strategy (First strategy) proposed in [15] is adopted; otherwise, the thread-adaptive allocation strategy with the nonconstant thread-group size (Second strategy) proposed in [16] is utilized. Given a matrix, should we use the first strategy or the second strategy? Here we present a selection method, whose main procedure is shown in Fig. 2.

Let us illustrate the selection method in Fig. 2 by apache2. For apache2, we have $n2max = 8$ and $n2avg = 6.74$. Obviously, $n2max$, $n2avg \in (2^2, 2^3]$, and $\alpha = \beta = 3$. Based on the selection method in Fig. 2, the first strategy is chosen.

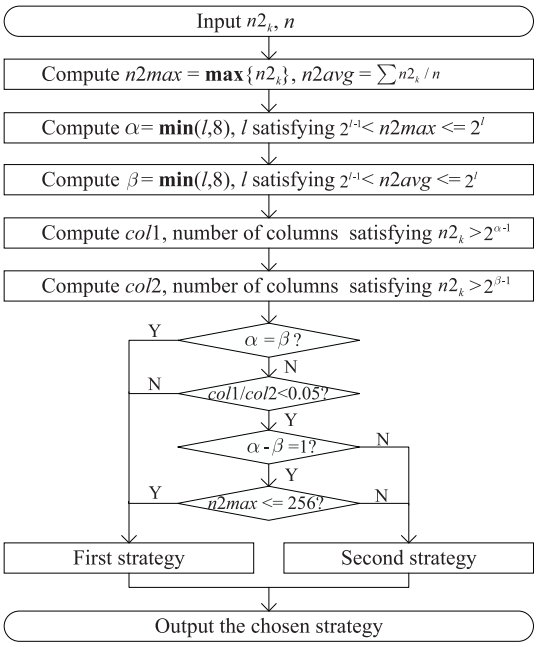


Fig. 2. Main procedure of selecting the First/Second strategy.

2.1. Pre-GSPAI stage

First, we compute the dimensions of all local submatrices. When computing m_k (one column of M), $k = 1, 2, \ldots, n$, the dimensions of the local submatrices ($n1_k$, $n2_k$) constructed for each column of the preconditioner are usually different. To simplify the accesses of data in the memory and enhance the coalescence, the dimensions of all local submatrices are uniformly defined as ($n1max$, $n2max$). Here $n1max = \max_k\{n1_k\}$ and $n2max = \max_k\{n2_k\}$.

Next, we choose whether to use the constant or nonconstant thread-group size for each column of the preconditioner. GSPAI-Opt fuses the advantages of SPAI-Adaptive [15] and GSPAI-Adaptive [16]. For SPAI-Adaptive, a thread-adaptive allocation strategy with the constant thread-group size is presented, and for GSPAI-Adaptive, a thread-adaptive allocation strategy with the nonconstant thread-group size is presented. For the convenience of readers, in the following contents, we introduce them respectively.

Thread-adaptive allocation strategy with the constant thread-group size: The optimized number of threads q is obtained by the following formula:

$$q = \min(2^s, nt), \tag{1}$$

s.t.

$$2^{s-1} < n2max \leqslant 2^s. \tag{2}$$

Here nt is the number of threads per block, and q threads are grouped into a thread group.

Thread-adaptive allocation strategy with the nonconstant thread-group size: First, for each $n2_k$, $k = 1, 2, \ldots, n$, the number of threads q_k assigned to the kth column of the preconditioner is computed by the following formula:

$$q_k = \min(2^s, nt), \tag{3}$$

s.t.

$$2^{s-1} < n2_k \leqslant 2^s. \tag{4}$$

Second, all q_k values are sorted in descending order. Finally, the thread-group size of each block is assigned by the procedure shown in Fig. 3.

Input: q, nt, n
Output: $WSize$, $BCol$, $blocks$
01. $i \leftarrow 0$; $blocks \leftarrow 0$; $BCol[0] \leftarrow 0$;
02. **while** $i < n$
03. $WSize[blocks] \leftarrow q[i]$;
04. i += $nt/WSize[blocks]$;
05. $blocks$++;
06. $BCol[blocks] \leftarrow i$;
07. **end while**

Fig. 3. Main procedure of assigning the thread-group size.

Table 1
Arrays used in GSPAI-Opt.

Array	Size	Type	Array	Size	Type
AData	nonzeros	double	$\widehat{m}$	ns × *n2max*	double
AIndex	nonzeros	integer	$\widehat{A}$	ns × *n1max* × *n2max*	double
APtr	*n*	integer	*R*	ns × *n2max* × *n2max*	double
RCol	*n*	integer	*I*	ns × *n1max*	integer
atomic	*n*	integer	*iPTR*	ns	integer
WSize	*blocks*	integer	*J*	ns × *n2max*	integer
BCol	*blocks*	integer	*jPTR*	ns	integer

Finally, we allocate global memory for arrays in Table 1, and *RCol*, *BCol*, and *WSize* values are transferred to the GPU global memory if the second strategy is applied.

2.2. Compute-GSPAI stage

Finding indices: This part is to find indices J and I by the constant/nonconstant thread-group size.

(1) ***Finding J and I by the constant thread-group size***: In this case, the thread-group size that is used to find J and I is same in all blocks. For the kernel that finds J, the threads inside each thread group read one column of the sparsity pattern M in parallel and store them to one subset of J. And then on this basis of J, we implement the construction of I. We establish a decision tree to find I based on the GPU feature parameters. Utilizing the decision tree, an optimized kernel for finding I is obtained for any given $n2max$ and $n1max$. Assume that the threads per block are 256 and NIVIDA GTX1070 GPU is used, Fig. 4 shows a segment of the decision tree for finding I. Here $sharedSize$ = number of columns of the preconditioner computed in a thread block × upper boundary closest to $n1max$. For example, when $n1max \leqslant 8$, $sharedSize = 32 \times 8$ and cuFindIBySharedMemory kernel with shared memory of 256 size is used. In the cuFindIBySharedMemory kernel, each thread group finds one subset of I, e.g., I_k, which mainly includes the following steps. First, the threads in the thread group load the row indices of the first column referenced in one subset of J, e.g., J_k, to shared memory sI. Second, the index vectors of successive columns referenced by J_k are compared in parallel with values in sI and new indices are appended to sI by utilizing the atomic operations. Third, inside the thread group, the indices of sI are sorted in ascending order in parallel. Finally, the indices of sI are copied to I_k. cuFindI kernel is similar to cuFindIBySharedMemory kernel except that the operations are executed on global memory instead of shared memory.

(2) ***Finding J and I by the nonconstant thread-group size***: The thread-group size of finding J and I is same in a block while it is usually different for different blocks. For the kernel that finds J, the threads inside each thread group read one column of the sparsity pattern M in parallel and store them to one subset of J. The main procedure of the kernel that finds I is as same as that in [16]. Each thread group is assigned to find one subset of I, e.g., I_k, which includes the following three stages. In the first stage, the thread group obtains the thread-group size $warpSize$. In the second stage, the row indices of the first column referenced in J_k are first loaded into I_k, and the row index vectors of successive columns that are referenced by J_k are calculated in parallel with values in I_k, and the new indices are appended to I_k by utilizing the atomic operations. In the third stage, the indices in I_k are sorted in ascending order in parallel.

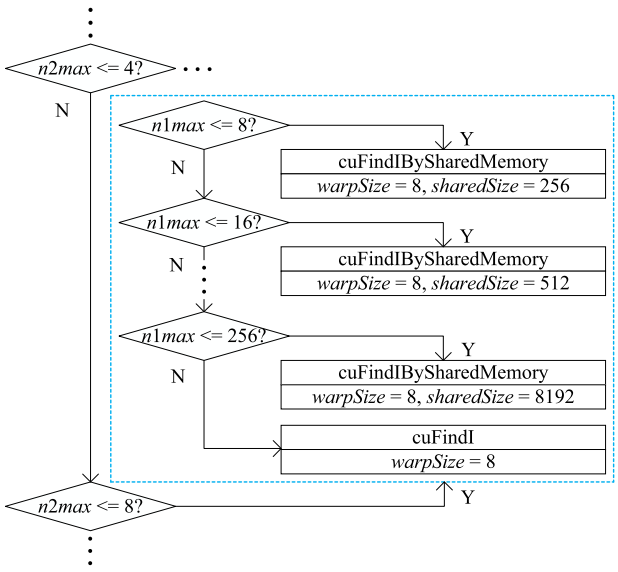


Fig. 4. A segment of the decision tree of using constant threads to find I.

Constructing the local submatrix: Using J and I obtained above, the local submatrix set, $\widehat{A}$, is computed by the constant/nonconstant thread-group size.

(1) ***Constructing the local submatrix by the constant thread-group size***: Each thread group is assigned to compute one subset of $\widehat{A}$, e.g., $\widehat{A}_k$, and all thread groups are the same size. Based on the GPU feature parameters, we establish a decision tree for constructing $\widehat{A}$. For any given $n2max$ and $n1max$, an optimized kernel for constructing $\widehat{A}$ is achieved by using the decision tree. For example, on NIVIDA GTX1070 GPU, assume that the threads per block are 256, Fig. 5 shows a segment of the decision tree for constructing $\widehat{A}$. When $4 < n2max \leqslant 8$, corresponding to different $n1max$, cuComputeTildeABySharedMemory kernel with shared memory of $sharedSize$ size and cuComputeTildeA kernel with non shared memory are selected. The main procedure of cuComputeTildeABySharedMemory kernel is listed as follows. For the thread group that calculates $\widehat{A}_k$, all threads in the thread group first read values in I_k into shared memory sI in parallel, and $\widehat{A}_k$ is established on the global memory by loading columns that are indexed by J_k and matching them to sI in parallel. cuComputeTildeA kernel is similar to cuComputeTildeABySharedMemory kernel except that I is executed on global memory instead of shared memory.

(2) ***Constructing the local submatrix by the nonconstant thread-group size***: In this case, each thread group is assigned to calculate one subset of $\widehat{A}$, e.g., $\widehat{A}_k$, and the thread-group size is same in a block but it can be different for different block. The main procedure of the kernel that constructs $\widehat{A}$ is as same as that in [16].

Decomposing the local submatrix into QR: This part is used to decompose the local submatrix into QR by the constant/nonconstant thread-group size.

(1) ***Decomposing the local submatrix into QR by the constant thread-group size***: The thread-group size of decomposing the local submatrix into QR is same in all blocks. Based on the GPU feature parameters, we establish a decision tree for decomposing the local submatrix into QR. For example, on NIVIDA GTX1070 GPU, assume that the threads per block are 256, Fig. 6 shows a segment of the decision tree for decomposing the local submatrix into QR. When $4 < n2max \leqslant 8$, two shared memories $sharedR$ and $sharedQ$ are

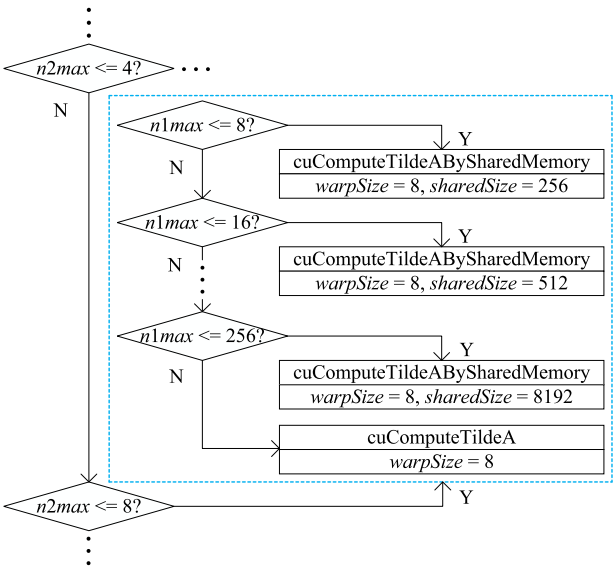


Fig. 5. A segment of the decision tree of using constant threads to construct $\hat{A}$.

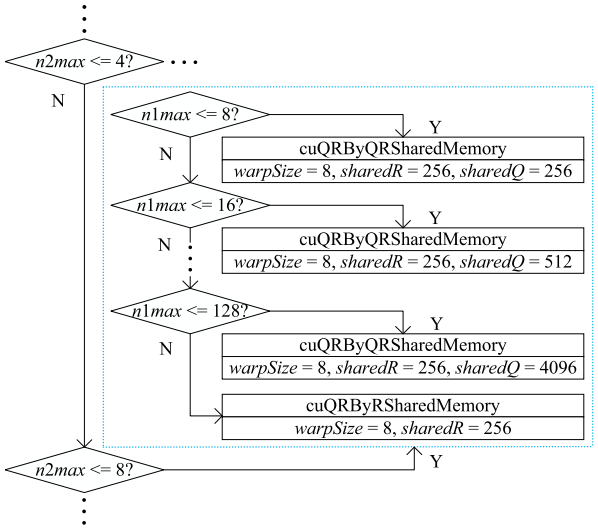


Fig. 6. A segment of the decision tree of using constant threads to decompose the local submatrix into QR.

used in the optimized kernel. Here the size of $sharedQ$ is related to $n1max$. In the cuQRByQRSharedMemory kernel, each thread group is responsible for one QR decomposition. In a thread group, the local submatrix, e.g., $\hat{A}_k$, is decomposed into QR by the following four steps at each iteration i. In the first step, the threads read the ith column of Q_k into shared memory sQ in parallel. In the second step, the ith row of the upper triangle matrix R_k are computed in parallel and are put into shared memory sR. In the third step, the column i of Q_k and sQ are concurrently normalized, and the projection factors R_k and sR are calculated. In the fourth step, the values of all columns of Q_k are updated by using shared memory sQ and sR in parallel. cuQRByRSharedMemory kernel is similar to cuQRByQRSharedMemory kernel except the shared memory sQ is not utilized.

(2) *Decomposing the local submatrix into QR by the nonconstant thread-group size*: The thread-group size of decomposing the local submatrix into QR is same in a block while it is usually different for different blocks. We establish a decision tree for decomposing the local submatrix into QR. For example, on NIVIDA GTX1070 GPU, the decision tree for decomposing the local submatrix into QR is shown

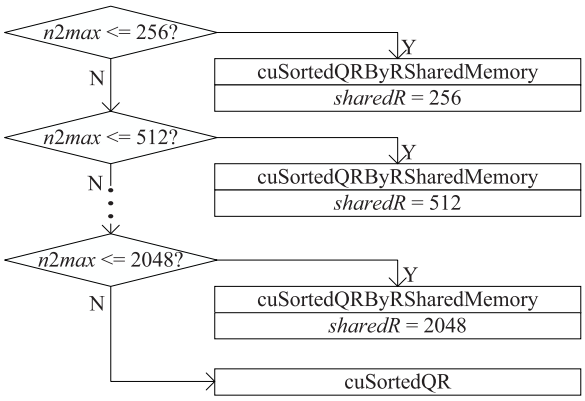


Fig. 7. Decision tree of using nonconstant threads to decompose the local submatrix into QR.

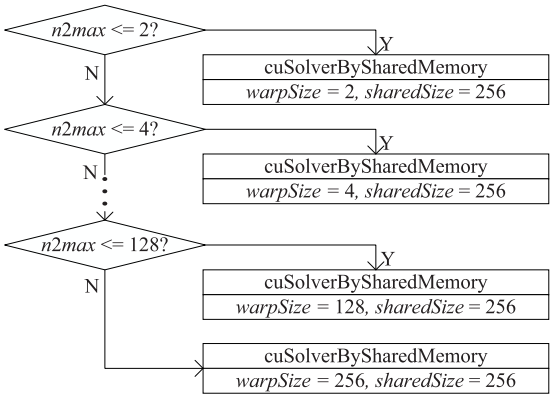


Fig. 8. Decision tree of using constant threads to solve the upper triangular linear system.

in Fig. 7 when the threads per block are 256. Obviously, utilizing the decision tree, an optimized kernel cuSortedQRByRSharedMemory corresponding to shared memory of $sharedR$ size or cuSortedQR kernel is chosen for a given $n2max$ value. The main procedure of cuSortedQRByRSharedMemory kernel is as same as that in [16]. cuSortedQR kernel is similar to cuSortedQRByRSharedMemory kernel except that the shared memory sR is not utilized.

Solving the upper triangular linear system: The values of $\hat{m}_k = R_k^{-1} Q_k^T \hat{e}_k$ are computed by the constant/nonconstant thread-group size.

(1) *Solving the upper triangular linear system by the constant thread-group size*: Each thread group computes one subset of $\hat{m}$ by solving an upper triangular linear system, and the thread-group size is same in all blocks. In this case, assume that the threads per block are 256, the decision tree for solving the upper triangular linear system is shown in Fig. 8. For any given $n2max$ value, an optimized kernel, cuSolverBySharedMemory with shared memory of 256 size and thread-group size of $warpSize$, is chosen. In the cuSolverBySharedMemory kernel, each thread group calculates a subset of $\hat{m}$, e.g., $\hat{m}_k$, and its procedure includes two steps. First, Calculate $Q_k^T \hat{e}_k$ in parallel and save the result to the shared memory xE. Second, the values of $\hat{m}_k$ are obtained by solving the upper triangular linear system $R_k \hat{m}_k = xE$, in parallel.

(2) *Solving the upper triangular linear system by the nonconstant thread-group size*: Each thread group is responsible for obtaining a subset of $\hat{m}$ by solving an upper triangular linear system, and the thread-group size is same inside a block but it can be different for different blocks. A decision tree is established to solve the upper triangular linear system. For example, Fig. 9 lists the decision tree for solving the upper triangular linear system on NIVIDA GTX1070 GPU. For any

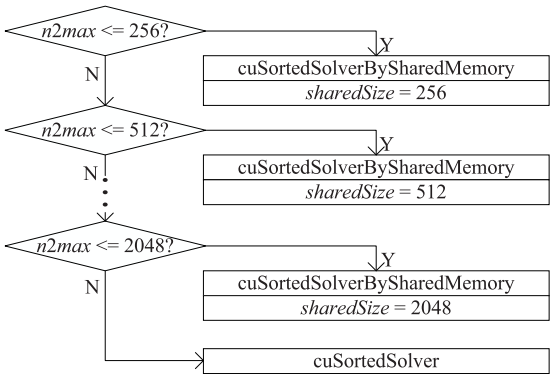


Fig. 9. Decision tree of using nonconstant threads to solve the upper triangular linear system.

Table 2
Overview of GPUs.

Hardware	GTX1070	A40
Cores	1920	10752
Clock speed (GHz)	1.506	1.305
Memory type	GDDR5	GDDR6
Memory size (GB)	8	48
Max-bandwidth (GB/s)	256	696
Compute capability	6.1	8.6

given $n2max$ value, we always choose an optimized kernel, which may be a cuSortedSolverBySharedMemory kernel that uses shared memory of $sharedSize$ size, or a cuSortedSolver kernel. The main procedure of cuSortedSolverBySharedMemory kernel is as same as that in [16]. cuSortedSolver kernel is similar to cuSortedSolverBySharedMemory kernel except that the shared memory xE is not used.

2.3. Post-GSPAI stage

In the *Post-GSPAI* stage, the preconditioner M is assembled in the CSC storage format which contains three arrays of $MPtr$, $MIndex$ and $MData$. Fig. 10 illustrates the procedure of assembling these arrays. First, *MPtr* is assembled utilizing *jPTR*. Second, $MData$ and $MIndex$ are assembled using $\hat{m}$ and J. In order to reduce the cost of array transfer, we assemble all arrays mentioned above on the GPU memory, and each thread group is responsible for generating one $\hat{m}_k$ to $MData$ and one J_k to $MIndex$.

3. Experimental results

In this section, we take two NVIDIA GPUs (GTX1070 and A40) shown in Table 2 to evaluate the performance of GSPAI-Opt. The test matrices are listed in Table 3, which are chosen from the SuiteSparse Matrix Collection [17], and have been widely used in some publications [14–16]. Table 3 summarizes the information of the sparse matrices, including the name, kind, number of rows, total number of nonzeros, average number of nonzeros, maximum number of nonzero entries

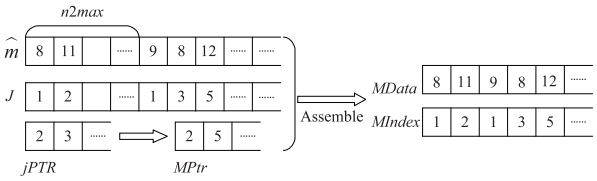


Fig. 10. Assemble M.

of columns, and minimum number of nonzero entries of columns. The matrices in Table 3 are chosen due to the following reasons. The matrices such as cbuckle, ASIC_320ks, power9, and Fault_639 are chosen to test whether the second strategy is chosen when the nonzero number of each column of the preconditioner has significant difference ($\alpha - \beta \geqslant 3$). The matrices such as 2cubes_sphere, offshore, apache2, G3_circuit are chosen to test whether the first strategy is chosen when the difference in the nonzero number of each column of the preconditioner is small ($\alpha = \beta$). The matrices such as msdoor and thermal2 are chosen to test whether the predicted strategy is well matched with the measured strategy when the difference in the nonzero number of each column of the preconditioner is large but not significant ($1 \leqslant \alpha - \beta < 3$). The source codes are compiled and executed using the CUDA toolkit 11.1 [18].

3.1. Accuracy of selection method

We take GTX1070 to test the accuracy of the proposed selection method of using the first strategy (denoted by S1) or the second strategy (denoted by S2). The sparse pattern of the preconditioner is a priori, so we test the accuracy in two popular patterns [14–16], $(E + |A|)^k, k = 1, 2$. The matrices in Table 3 are used as the test matrices. For all test matrices, both the optimal strategy predicted by the proposed selection method and the strategy obtained from actual tests are shown in Table 4. Note that if $|t1 - t2| / \max(t1, t2) \leqslant 0.05$, both S1 and S2 can be considered as the measured optimization strategy; otherwise, the strategy corresponding to $\min(t1, t2)$ is chosen as the measured optimization one. Here $t1$ and $t2$ are the time of constructing the preconditioner using S1 and S2, respectively. We can observe that for the two sparsity patterns, the estimated and measured optimal strategies are matched very well for the test cases. This verifies good accuracy of our proposed selection method.

3.2. Performance comparison

In order to test the effectiveness of our proposed GSPAI-Opt, we take the sparsity pattern $(E + |A|)$ to compare it with a static SPAI preconditioning algorithm in ViennaCL (denoted by SSPAI-VCL) [13], and two recent SPAI preconditioning algorithms SPAI-Adaptive [15] and GSPAI-Adaptive [16] on GTX1070 and A40, and their comparison results are listed in Tables 5 and 6, respectively. Moreover, since SSPAI-VCL cannot be suitable for the sparsity pattern $(E + |A|)^2$, only the comparison results of SPAI-Adaptive, GSPAI-Adaptive and GSPAI-Opt on two GPUs are shown in Table 7. In each table, for any matrix and

Table 3
Descriptions of test matrices.

Name	Kind	Rows	Nonzeros	Avg	Max	Min
cbuckle	Structural	13,681	676,515	49.45	600	26
2cubes_sphere	Electromagnetics	101,492	1,647,264	16.23	31	5
offshore	Electromagnetics	259,789	4,242,673	16.33	31	5
ASIC_320ks	Circuit simulation	321,671	1,316,085	4.09	210	1
apache2	Structural	715,176	4,817,870	6.74	8	4
G3_circuit	Circuit simulation	1,585,478	7,660,826	4.83	6	2
power9	Semiconductor device	155,376	1,887,730	12.15	627	1
msdoor	Structural	415,863	19,173,163	46.10	77	1
thermal2	Thermal	1,228,045	8,580,313	6.99	11	1
Fault_639	Structural	638,802	27,245,944	42.65	267	1

Table 4
Predicted and measured sparsity pattern.

Matrix	$(E+\|A\|)$		$(E+\|A\|)^2$	
	Predicted	Measured	Predicted	Measured
cbuckle	S2	S2	S2	S2
2cubes_sphere	S1	S1	S1	S1/S2
offshore	S1	S1	S1	S1/S2
ASIC_320ks	S2	S2	S2	S2
apache2	S1	S1	S1	S1
G3_circuit	S1	S1	S1	S1
power9	S2	S2	S2	S2
msdoor	S1	S1/S2	S1	S1/S2
thermal2	S1	S1	S1	S1
Fault_639	S2	S2	S2	S2

Table 5
Comparison of four algorithms with $(E+|A|)$ on GTX1070.

Matrix	SSPAI-V	SPAI-A	GSPAI-A	GSPAI-Opt
cbuckle	N/A	7.976	2.046	1.815
	N/A	0.362	0.356	0.348
	N/A	96	96	96
	N/A	8.338	2.402	**2.163**
2cubes_sphere	7.278	0.833	0.697	0.539
	0.025	0.300	0.296	0.294
	5	4	4	4
	7.303	1.133	0.993	**0.833**
offshore	20.468	2.177	2.052	1.380
	0.053	0.323	0.324	0.327
	12	5	5	5
	20.521	2.500	2.376	**1.707**
ASIC_320ks	N/A	5.000	1.398	0.846
	N/A	0.347	0.342	0.339
	N/A	10	10	10
	N/A	5.347	1.740	**1.185**
apache2	5.722	0.238	0.328	0.222
	7.963	3.583	3.574	3.585
	2503	1090	1090	1090
	13.685	3.821	3.902	**3.807**
G3_circuit	/	0.148	0.170	0.148
	/	2.887	2.881	2.885
	>10 000	468	468	468
	/	3.035	3.051	**3.033**
power9	N/A	4.504	10.620	2.848
	N/A	0.436	0.435	0.418
	N/A	37	37	37
	N/A	4.940	11.055	**3.266**
msdoor	N/A	59.794	21.374	20.206
	N/A	5.378	5.373	5.442
	N/A	892	892	892
	N/A	65.172	26.747	**25.648**
thermal2	/	0.401	0.527	0.340
	/	11.700	11.696	11.701
	>10 000	2086	2086	2086
	/	12.101	12.223	**12.041**
Fault_639	N/A	185.893	42.994	37.524
	N/A	10.032	10.022	10.013
	N/A	1226	1226	1226
	N/A	195.925	53.016	**47.537**

any given preconditioner, the first two rows are the execution time of the preconditioning algorithm and GPUPBICGSTAB, respectively, and the third row is the iteration number of GPUPBICGSTAB, and the fourth row is the total of the first two rows; if the iteration number of GPUPBICGSTAB is more than 10,000, we record the number of iterations ">10 000" in the third row, and the other rows that record the time are represented by "/"; if the out-of-memory error for GPUPBICGSTAB is encountered, all rows will be denoted by "N/A". The time unit is second (s), and the minimum value of the fourth row for each matrix is marked in the red font. For the convenience, SSPAI-VCL + GPUPBICGSTAB, SPAI-Adaptive + GPUPBICGSTAB, GSPAI-Adaptive + GPUPBICGSTAB and GSPAI-Opt + GPUPBICGSTAB are denoted by SSPAI-V, SPAI-A, GSPAI-A and GSPAI-Opt, respectively.

Table 6
Comparison of four algorithms with $(E+|A|)$ on A40.

Matrix	SSPAI-V	SPAI-A	GSPAI-A	GSPAI-Opt
cbuckle	N/A	3.717	1.167	0.878
	N/A	0.272	0.318	0.236
	N/A	96	96	96
	N/A	3.989	1.485	**1.114**
2cubes_sphere	4.952	0.336	0.327	0.221
	0.019	0.236	0.311	0.317
	5	4	4	4
	4.971	0.572	0.638	**0.538**
offshore	13.726	0.858	1.075	0.627
	0.047	0.255	0.284	0.269
	12	5	5	5
	13.773	1.113	1.359	**0.896**
ASIC_320ks	N/A	2.764	0.772	0.307
	N/A	0.253	0.316	0.286
	N/A	10	10	10
	N/A	3.017	1.088	**0.593**
apache2	4.471	0.122	0.201	0.088
	1.646	1.353	1.709	1.331
	2503	1256	1256	1256
	6.117	1.475	1.910	**1.419**
G3_circuit	/	0.069	0.106	0.061
	/	1.072	1.189	1.068
	>10 000	472	472	472
	/	1.141	1.295	**1.129**
power9	N/A	2.073	7.346	1.519
	N/A	0.336	0.359	0.345
	N/A	37	37	37
	N/A	2.409	7.705	**1.864**
msdoor	N/A	20.142	9.964	8.225
	N/A	1.635	2.033	1.790
	N/A	656	656	656
	N/A	21.777	11.997	**10.015**
thermal2	/	0.176	0.273	0.144
	/	3.757	3.806	3.699
	>10 000	2186	2186	2186
	/	3.933	4.079	**3.843**
Fault_639	N/A	65.339	20.396	15.558
	N/A	3.348	3.821	3.419
	N/A	1149	1149	1149
	N/A	68.687	24.217	**18.977**

From Tables 5 and 6, we can see that as compared to SSPAI-VCL on two GPUs, for some matrices such as thermal2 and G3_circuit, GPUPBICGSTAB with SSPAI-VCL cannot converge in 10,000 iterations while GSPAI-Opt can. Especially, for the matrices with large $n2max$ value, e.g., cbuckle, power9, ASIC_320ks, msdoor and Fault_639, GPUPBICGSTAB with SSPAI-VCL will encounter the out-of-memory error. Furthermore, for 2cubes_sphere, offshore, and apache2, the total time of SSPAI-VCL and GPUPBICGSTAB on two GPUs is much more than that of GSPAI-Opt and GPUPBICGSTAB. This verifies that GSPAI-Opt is much better than SSPAI-VCL for all test matrices. Compared with SPAI-Adaptive and GSPAI-Adaptive, GSPAI-Opt does not only have smaller execution time, but also the total time of GSPAI-Opt and GPUPBICGSTAB is much less than that of SPAI-Adaptive and GPUPBICGSTAB and that of GSPAI-Adaptive and GPUPBICGSTAB for all test cases. Fig. 11 shows the execution time ratios of SPAI-Adaptive to GSPAI-Opt and GSPAI-Adaptive to GSPAI-Opt on two GPUs. On the GTX1070 GPU, the minimum and maximum execution time ratios of SPAI-Adaptive to GSPAI-Opt are roughly 1.0 and 4.95, respectively, and the average ratio is roughly 2.62; the minimum and maximum execution time ratios of GSPAI-Adaptive to GSPAI-Opt are roughly 1.13 and 3.73, respectively, and the average ratio is roughly 1.57. On the A40 GPU, the minimum and maximum execution time ratios of SPAI-Adaptive to GSPAI-Opt

Table 7
Comparison of three algorithms with $(E+|A|)^2$.

Matrix	GTX1070			TITANXp		
	SPAI-A	GSPAI-A	GSPAI-Opt	SPAI-A	GSPAI-A	GSPAI-Opt
cbuckle	29.119	9.213	7.159	12.773	4.361	3.071
	0.492	0.470	0.477	0.272	0.319	0.270
	66	66	66	55	55	55
	29.479	9.564	**7.500**	13.045	4.680	**3.341**
2cubes_sphere	50.005	25.548	25.114	16.383	10.040	8.461
	0.171	0.171	0.161	0.132	0.141	0.151
	2	2	2	4	4	4
	50.176	25.719	**25.275**	16.515	10.181	**8.612**
offshore	133.587	73.907	69.016	44.813	30.238	22.370
	0.213	0.213	0.196	0.173	0.186	0.207
	3	3	3	3	3	3
	133.800	74.120	**69.212**	44.986	30.424	**22.577**
ASIC_320ks	10.223	2.460	1.699	5.667	1.185	1.040
	0.346	0.338	0.341	0.287	0.282	0.291
	6	6	6	6	6	6
	10.569	2.798	**2.040**	5.954	1.467	**1.331**
apache2	3.934	3.627	3.249	1.500	1.391	1.314
	2.913	2.907	2.883	1.026	0.989	0.984
	629	629	629	600	600	600
	6.847	6.534	**6.132**	2.526	2.380	**2.298**
G3_circuit	1.864	2.094	1.467	0.709	0.946	0.652
	2.291	2.283	2.292	0.985	0.946	0.982
	299	299	299	345	345	345
	4.155	4.377	**3.759**	1.714	1.892	**1.634**
power9	20.575	26.960	13.497	10.866	17.445	6.981
	0.411	0.409	0.393	0.329	0.302	0.309
	21	21	21	21	21	21
	20.986	27.369	**13.890**	11.195	17.747	**7.290**
msdoor	335.084	115.082	108.323	108.962	44.192	33.335
	2.471	2.470	2.471	1.072	0.959	0.955
	892	892	892	298	298	298
	337.555	117.552	**110.794**	110.034	45.151	**33.290**
thermal2	4.082	3.753	2.991	1.526	1.431	1.260
	11.462	11.466	11.469	3.469	3.477	3.529
	1502	1502	1502	1464	1464	1464
	15.544	15.219	**14.460**	4.995	4.908	**4.789**
Fault_639	627.18	201.616	180.840	235.209	83.214	63.198
	16.245	6.242	6.242	2.373	1.748	1.751
	588	588	588	473	473	473
	633.425	207.858	**187.082**	237.582	84.962	**64.949**

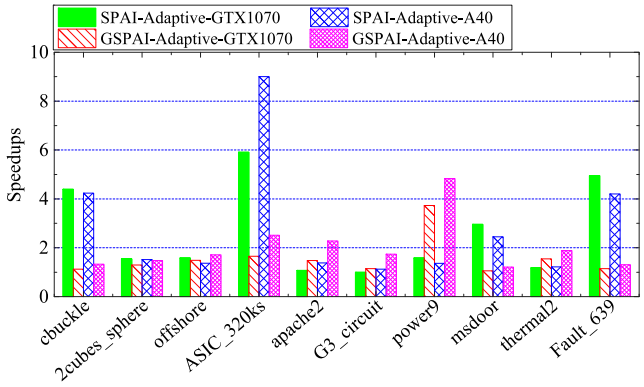


Fig. 11. Execution time ratios of SPAI-Adaptive vs GSPAI-Opt and GSPAI-Adaptive vs GSPAI-Opt for $E+|A|$ on two GPUs.

are roughly 1.12 and 9, respectively, and the average ratio is roughly 2.79; the minimum and maximum execution time ratios of GSPAI-Adaptive to GSPAI-Opt are roughly 1.21 and 4.83, respectively, and the average ratio is roughly 2.03. These observations verify that GSPAI-Opt outperforms SPAI-Adaptive and GSPAI-Adaptive.

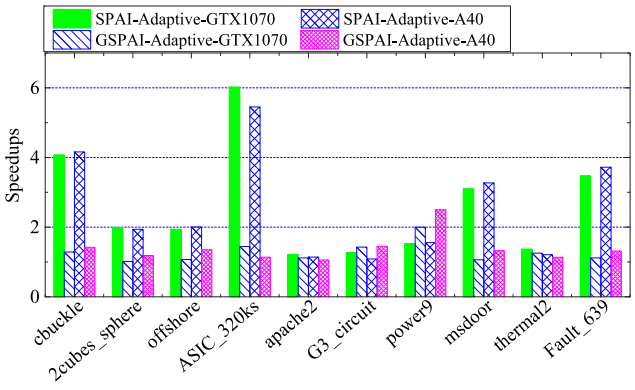


Fig. 12. Execution time ratios of SPAI-Adaptive vs GSPAI-Opt and GSPAI-Adaptive vs GSPAI-Opt for $(E+|A|)^2$ on two GPUs.

For the sparsity pattern of $(E+|A|)^2$, from Table 7, we can observe that comparing with SPAI-Adaptive and GSPAI-Adaptive, we can draw the same conclusion as the sparsity pattern of $(E+|A|)$ for GSPAI-Opt. GSPAI-Opt is much better than SPAI-Adaptive and GSPAI-Adaptive. This can also be confirmed from Fig. 12. On the GTX1070 GPU, the minimum and maximum execution time ratios of SPAI-Adaptive to GSPAI-Opt are roughly 1.99 and 6.02, respectively, and the average ratio is roughly 2.59; the minimum and maximum execution time ratios of GSPAI-Adaptive to GSPAI-Opt are roughly 1.01 and 2, respectively, and the average ratio is roughly 1.28. On the A40 GPU, the minimum and maximum execution time ratios of SPAI-Adaptive to GSPAI-Opt are roughly 1.14 and 5.45, respectively, and the average ratio is roughly 2.55; the minimum and maximum execution time ratios of GSPAI-Adaptive to GSPAI-Opt are roughly 1.05 and 2.5, respectively, and the average ratio is roughly 1.39.

4. Conclusion

In this study, we propose an optimized sparse approximate inverse preconditioners on GPU called GSPAI-Opt. In the proposed GSPAI-Opt, for any given sparsity pattern of the preconditioner, a selection strategy is presented to determine the size of the thread group for each column of the preconditioner. Furthermore, no matter which strategy we choose, each column of the preconditioner is performed in parallel within a thread group. The experimental results verify that GSPAI-Opt can well fuse the advantages of SPAI-Adaptive and GSPAI-Adaptive and is highly effective.

Next, we will further do research in this field, and apply the proposed GSPAI-Opt to more practical problems.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] CUDA C Programming guide 1.0, 2007, https://developer.nvidia.com/content/cuda-10.
[2] X. Chu, J. Gao, B. Sheng, Efficient concurrent L1-minimization solvers on GPUs, Comput. Syst. Sci. Eng. 38 (3) (2021) 305–320.
[3] J. Gao, Y. Xia, R. Yin, G. He, Adaptive diagonal sparse matrixvector multiplication on GPU, J. Parallel Distrib. Comput. 157 (2021) 287–302.
[4] K. Li, W. Yang, K. Li, A hybrid parallel solving algorithm on GPU for quasi-tridiagonal system of linear equations, IEEE Trans. Parallel Distrib. 27 (10) (2016) 2795–2808.
[5] S.C. Rennich, D. Stosic, T.A. Davis, Accelerating sparse cholesky factorization on GPUs, Parallel Comput. 59 (2016) 140–150.

[6] H. Anzt, M. Gates, J. Dongarra, M. Kreutzer, G. Wellein, M. Kohler, Preconditioned Krylov solvers on GPUs, Parallel Comput. 68 (2017) 32–44.
[7] E. Chow, A. Patel, Fine-grained parallel incomplete LU factorization, SIAM J. Sci. Comput. 37 (2) (2015) C169–C193.
[8] J. Gao, X. Chu, X. Wu, J. Wang, G. He, Parallel dynamic sparse approximate inverse preconditioning algorithm on GPU, IEEE Trans. Parallel Distrib. 33 (12) (2022) 4723–4737.
[9] Y. Saad, M.H. Schultz, GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM J. Sci. Stat. Comput. 7 (3) (1986) 856–869.
[10] H.A. van der Vorst, Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of non-symmetirc linear systems, SIAM J. Sci. Stat. Comput. 12 (3) (1992) 631–644.
[11] E. Chow, A priori sparsity patterns for parallel sparse approximate inverse preconditioners, SIAM J. Sci. Comput. 21 (5) (2000) 1804–1822.
[12] J. Gao, K. Wu, Y. Wang, P. Qi, G. He, GPU-accelerated preconditioned GMRES method for two-dimensional Maxwell's equations, Int. J. Comput. Math. 94 (10) (2017) 2122–2144.
[13] K. Rupp, R. Tillet, F. Rudolf, et al., ViennaCL-linear algebra library for multi- and many-core architectures, SIAM J. Sci. Comput. 38 (5) (2016) S412–S439.
[14] M.M. Dehnavi, D.M. Fernandez, J.L. Gaudiot, Parallel sparse approximate inverse preconditioning on graphic processing units, IEEE Trans. Parallel Distrib. 24 (9) (2013) 1852–1861.
[15] G. He, R. Yin, J. Gao, An efficient sparse approximate inverse preconditioning algorithm on GPU, Concurr. Comput.-Pract. Exp. 32 (7) (2020) e5598, http://dx.doi.org/10.1002/cpe.5598.
[16] J. Gao, Q. Chen, G. He, A thread-adaptive sparse approximate inverse preconditioning algorithm on multi-GPUs, Parallel Comput. 101 (2021) 102724, http://dx.doi.org/10.1016/j.parco.2020.102724.
[17] T.A. Davis, Y. Hu, The university of florida sparse matrix collection, ACM Trans. Math. Software 38 (1) (2011) 1–25.
[18] CUDA C Programming guide 11.1, 2021, http://docs.nvidia.com/cuda/cuda-c-programming-guide.

BenchCouncil Transactions on Benchmarks, Standards and Evaluations 2 (2022) 100090

Contents lists available at ScienceDirect

BenchCouncil Transactions on Benchmarks, Standards and Evaluations

journal homepage: www.keaipublishing.com/en/journals/benchcouncil-transactions-on-benchmarks-standards-and-evaluations/

Research Article

Performance characterization and optimization of pruning patterns for sparse DNN inference

Yunjie Liu, Jingwei Sun *, Jiaqiang Liu, Guangzhong Sun

University of Science and Technology of China, Hefei, China

ARTICLE INFO

Keywords:
Deep neural network
Pruning
Matrix multiplication

ABSTRACT

Deep neural networks are suffering from over parameterized high storage and high consumption problems. Pruning can effectively reduce storage and computation costs of deep neural networks by eliminating their redundant parameters. In existing pruning methods, filter pruning achieves more efficient inference, while element-wise pruning maintains better accuracy. To make a trade-off between the two endpoints, a variety of pruning patterns has been proposed. This study analyzes the performance characteristics of sparse DNNs pruned by different patterns, including element-wise, vector-wise, block-wise, and group-wise. Based on the analysis, we propose an efficient implementation of group-wise sparse DNN inference, which can make better use of GPUs. Experimental results on VGG, ResNet, BERT and ViT show that our optimized group-wise pruning pattern achieves much lower inference latency on GPU than other sparse patterns and the existing group-wise pattern implementation.

1. Introduction

Deep neural networks (DNNs) have achieved remarkable performance in the field of artificial intelligence and have attracted the interest of many researchers. In recent years, deep neural networks have been widely applied in numerous applications, including computer vision [1], natural language processing [2], recommendation systems [3], etc.

In order to achieve high accuracy, DNNs usually have the property of over-parameterized. In other words, they contain redundant parameters that cost large storage and are difficult to be deployed to resource-constrained devices. The inference latency of DNNs is also affected due to the large amount of computational operations. To address this issue, researchers have proposed various methods to compress DNN models. Pruning is a representative and effective model compression method. It identifies and removes redundant parameters in a DNN according to specific criteria. Ideally, after conducting pruning method, the amount of both model parameters and computational operations is reduced, and the inference time cost should also be reduced.

In practice, pruning does not ensure efficient inference. According to the granularity of pruned parameters, existing pruning methods falls along a spectrum between unstructured pruning and structured pruning. When the unstructured element-wise pattern is used for pruning, more parameters can be pruned. However, the pruned model is unfriendly on commodity GPU architectures due to unaligned and non-coalescing data access [4–6]. In this case, although the number of model parameters and the number of computational operations are reduced, the inference time can be even higher than that of the dense model due to the imperfect hardware support for the sparse computation [7]. When pruning uses the structured filter pruning, the parallel computing ability of GPUs can be better utilized due to its regular computation. However with this method, the number of pruned parameters is limited, while the model accuracy may drops significantly. To make a trade-off between the two endpoints, a variety of pruning patterns has been proposed, such as vector-wise [8,9] and block-wise pruning [10,11]. Vector-wise pruning divides the parameters of each row into vectors with equal size, and prune equal proportions of the parameters in each vector. Block-wise pruning divides the weight matrix into matrix blocks of specific shapes and removes the redundant blocks according to importance criteria of each block. Compared with filter pruning, these structured pruning patterns reserve more regular, balanced, and partially dense non-zero elements. However, the resulting sparse models from these pruning patterns still require specific runtime support.

In this study, we analyze the performance characteristics of different sparse DNNs, including element-wise, vector-wise, block-wise, and group-wise patterns. We find that these pruning patterns with off-the-shelf sparse computing libraries (e.g., cuSPARSE) are difficult to make full use of GPU ability. We then propose an efficient implementation of structured sparse DNN inference based on group-wise pattern. More specifically, for convolutional neural networks (CNNs), group-wise pattern removes the parameters with the same indixes in

* Corresponding author.
E-mail addresses: lyj06@mail.ustc.edu.cn (Y. Liu), sunjw@ustc.edu.cn (J. Sun), jqliu42@mail.ustc.edu.cn (J. Liu), gzsun@ustc.edu.cn (G. Sun).

https://doi.org/10.1016/j.tbench.2023.100090
Received 7 December 2022; Received in revised form 5 March 2023; Accepted 5 March 2023
Available online 7 March 2023

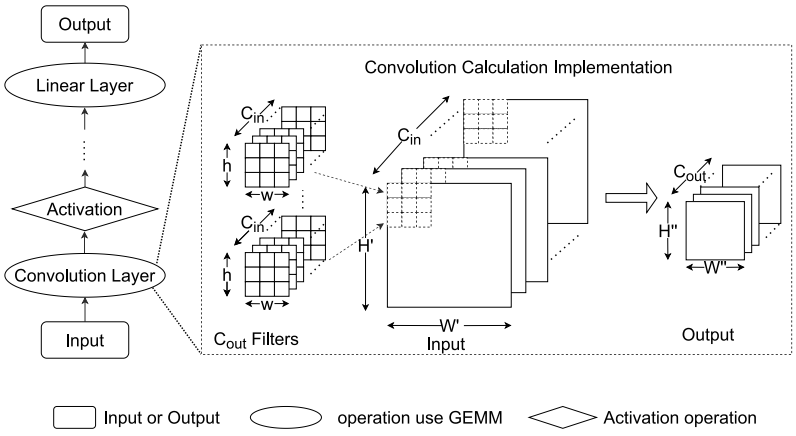


Fig. 1. Convolutional neural network.

all channels. For recurrent neural networks (RNNs) and transformer-based models, group-wise pattern removes rows of each weight matrix. Based on this pruning pattern, we convert the dominant computation kernels of pruned CNNs, RNNs, and Transformers to general matrix multiplication (GEMM) operations. Current deep learning programming frameworks (e.g., PyTorch, Tensorflow) and hardware platforms support well-developed GEMM operations. Therefore, our implementation can make better use of GPUs. Besides, group-wise pruning pattern only constrains the layout of non-zero elements. It is easy to combine the pattern with existing sophisticated pruning schedules and importance criteria, like Dynamic Sparse Training [12], Lottery Ticket Hypothesis [4,13], Magnitude [14,15], Taylor [16], Hessian [17], etc.

The main contributions of this paper are summarized as follows:

- We conduct an empirical study on existing mainstream fine-grained and structured pruning patterns. We compare their inference performance under varying conditions and indicate their inefficiency on GPU with off-the-shelf sparse computing library.
- We propose an efficient implementation of group-wise pruning pattern. The implementation converts group-wise sparse matrix-matrix multiplication into GEMM operations and optimizes the memory accesses according to GPU hardware characteristics. It makes full use of existing runtime libraries and GPU hardware support.

2. Background and related work

2.1. DNN model pruning

Generally, neural networks have over-parameterized property and contain redundant parameters. By analyzing and removing these redundant parameters during or after training, a neural network can be optimized to obtain a lower execution time and consume less memory resources when it is deployed to a target device. This process is the pruning of neural networks.

LeCun et al. in [18] pioneered the optimal brain damage (OBD) method that treats the individual weights as a unit. Hassibi et al. [19,20] proposed an optimal brain surgeon (OBS) method based on the optimal brain damage method with the addition of an update step based on the surgical recovery weights, based on the diagonal assumption, the extreme value assumption and the quadratic assumption. Later, Han et al. [21] proposed that learning only the important connections in the network can reduce the number of model parameters and computation without affecting the final accuracy of the network, and proposed the classical pruning-retraining framework. Li et al. [15] proposed a compression technique based on convolutional kernel pruning. Hu et al. [22] proposed to use both the base model output and the pruned classification loss function to supervise the channel selection at each layer, especially introducing additional losses to encode the difference between the features in the base model and the pruned model feature maps. By considering reconstruction error, additional loss and classification loss simultaneously, the accuracy of the pruned model is greatly improved.

2.2. Hardware-aware acceleration for pruned DNN models

Dense model. A convolutional neural networks mainly contains two layer types: convolutional layer and linear layer. The computation of linear layer can be simply regarded as matrix multiplication. The convolutional layer is shown in Fig. 1, which can be computed by converting the convolutional algorithm to matrix multiplication using im2col algorithm. For the BERT model and ViT model, its main computational part, encoders structure, can also be regarded as some column matrix multiplication, as shown in Fig. 2. Therefore, the key point of reducing the latency of a dense neural network model is to reduce the latency of matrix multiplication. Generalized matrix multiplication (GEMM) is used in deep learning to perform the above matrix operations. Since GEMM has been well-developed for a long time, most of the existing programming frameworks and commercial hardware can efficiently support dense model acceleration.

A model pruned by structured pruning, such as channel pruning or filter pruning, is also dense model, so it can be calculated by GEMM. Due to the strong constraint of the structured pruning pattern, the accuracy is usually worse than fine-grained pruning. Related studies focus on maintain higher accuracy. FlexPruner [23], a filter pruning method with flexible rate. It is based on a greedy strategy to select the filters to be pruned. Li et al. [24] extend the optimization space for pruning, so their method is able to compress the model more effectively. The MaskACC pruning method [25] dynamically reorganizes tensors and mask information used in convolutions to avoid unnecessary computations, so that the computational efficiency of the pruning process is improved.

Sparse model. GPUs are originally designed for dense linear algebra computation and are not ideal for sparse computations. Therefore, the design of pruning pattern is essential to the inference latency of pruned models. Zhu et al. [26] used a vector-wise pruning pattern to ensure a balanced workload for the pruned network. By adding a sparse mode with extended instruction set and hardware support, it can run on Tensor Core. Lin et al. [27] tiled the weights and divided them according to a similar vector-wise pattern to remove redundant whole vectors, which has a better trade-off between latency and performance compared to weight pruning and filter pruning. Anwar et al. [28] first explored kernel-level pruning, and proposed an intra-kernel strided pruning method, which prunes a sub-vector in a fixed stride. Guo et al. [29] proposed a Tile-wise mode, which first divides the matrix

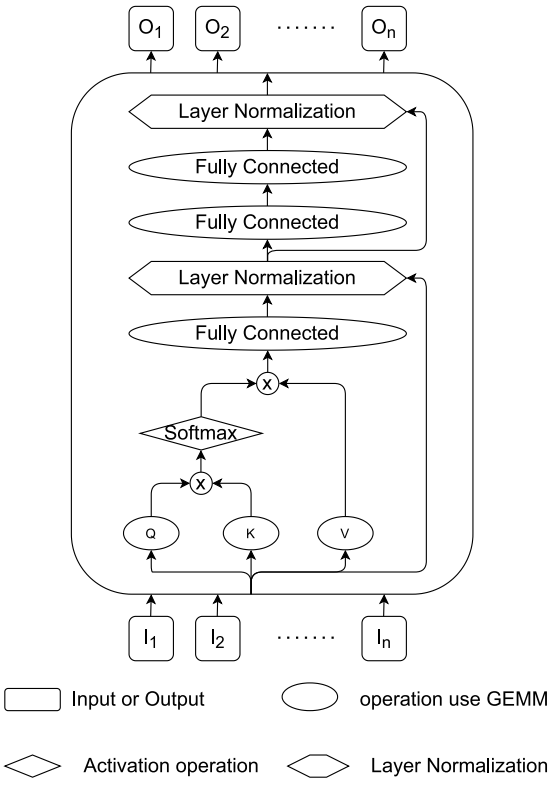


Fig. 2. Encoder structure in BERT model. Many of these operations use GEMM for completion.

into several larger Tile blocks according to the parallelism property during hardware computation, and prunes the ranks and columns within the blocks. Lebedev et al. proposed a group-wise pruning pattern in the convolutional layer in [30]. However, the pattern was combined with the Brain Damage criterion to propose a whole set of pruning method. We instead propose an efficient implementation of the group-wise pattern that focuses more on the inference time. Moreover, this work extends group-wise in the linear layer to achieve optimization of the whole neural network in terms of inference time.

In addition to innovations and research on pruning patterns, researchers also focus on efficient implementation and execution of pruned models. For instance, SparTA [31] is an end-to-end model sparsity framework that uses Tensor-with-Sparsity Attribute (TeSA) to build sparse models. Providing speedup for unstructured pruning and block-wise granularity pruning, it is compatible with a variety of sparse models and optimization techniques, facilitating sparse algorithms to explore better sparse models.

Compared with the existing works, the work proposed in this paper make better use of off-the-shelf dense computing libraries provided by vendors, e.g., cuBLAS. It has simpler implementation and higher portability. It avoids to use low-level APIs and hyper-parameters (e.g., tile width, block size) that are related to hardware architectures, so it can run on all NVIDIA GPUs, and achieve acceleration without specific tuning.

3. Performance characterization

3.1. Preliminary

The most intensive computation in a deep neural network mainly occurs in two layers: convolutional layer and linear layer. The computation of a convolutional layer transforms an input map U with C_{in} channels of size $W' \times H'$ into an output map V with C_{out} channels of size $W'' \times H''$. The relationship between the specific W', H', W'' and H'' is related to the padding and stride settings in the convolutional layer. The above transformation can be represented by the following formula:

$$V(c_o, x, y) = \sum_{c_i=1}^{C_{in}} \sum_{\substack{i=1\ldots h \\ j=1\ldots w}} K(c_o, c_i, i, j) \cdot U(c_i, x+i-\frac{h+1}{2}, y+j-\frac{w+1}{2}) \tag{1}$$

where K is a four-dimensional kernel tensor of size $C_{out}\times C_{in}\times h\times w$. The C_{out} corresponds to the output maps, the C_{in} corresponds to the input maps, and the h and w correspond to the convolutional kernel size.

To calculate (1), the kernel tensor K is reshaped into a two-dimensional weight matrix F with height $I' = C_{in}\times h\times w$ and width C_{out}. The input data U is reshaped into a two-dimensional input expansion matrix X with height $W'' \times H''$ and width $I' = C_{in} \times h \times w$. Each row consists of a square expansion that is computed with the corresponding convolutional kernel. Now we just need the following calculation [32]:

$$\tilde{V}(x', y') = \sum_{i=1}^{I'} X(x', i) * F(i, y') \tag{2}$$

The size of the matrix $\tilde{V}$ is $W''\times H''$ in height and C_{out} in width and it contains all the output data of the convolutional layer. The correct output V is obtained by reshaping $\tilde{V}$.

A linear layer transforms a tensor with F_{in} dimensions to a tensor with F_{out} dimensions. The transformation can be expressed using the following equation:

$$Y(f_o) = \sum_{f_i=1}^{F_{in}} X(f_i) * W(f_i, f_o) \tag{3}$$

Where W is a tensor with height F_{in} and width F_{out}.

Since the computation of both convolutional layer and linear layer can be converted to matrix multiplication, the operation of pruning a layer is just reducing the parameters in the two-dimensional weight expansion matrix F of a convolutional layer or the W matrix of a linear layer.

3.2. Pruning patterns

A pruning method mainly consists of three components: pruning pattern, pruning schedule, and pruning criterion. Pruning pattern defines the layout of reserved non-zero elements. Pruning schedule determines the occurrence time of pruning, such as pruning after training [33,34], during training [12,35], and before training [36,37]. Pruning criterion measures the importance of a set of parameters, determining whether these parameters are pruned [4,12–17]. The AI research community usually concerns more about the schedule and the criterion, which have critical impact on the model compression ratio and inference accuracy. In this study, we focus on pruning pattern, which is essential to the execution latency and hardware utilization of pruned DNNs on GPUs.

Fig. 3 shows examples of element-wise, vector-wise, block-wise, and group-wise patterns, respectively.

The element-wise pattern [4–6] is also known as unstructured weight pruning. After the kernel tensor has been reshaped into a two-dimensional weight expansion matrix, unimportant individual parameters are removed according to specific pruning criterion. It can remove a large portion of parameters, resulting in a significant reduction of the model size. However, the layout of parameters obtained by such a pruning pattern is irregular and does not substantially help improve the execution performance of sparse DNN inference.

The vector-wise pattern [8,9] retains more local structure compared to element-wise. Vector-wise pattern divides the weight expansion matrix into vectors of equal size. For example, if the weight expansion

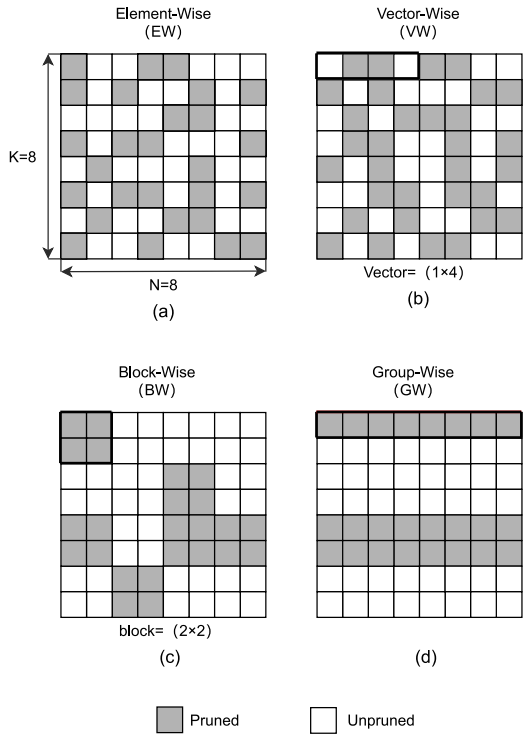


Fig. 3. Comparison of different pruning patterns. M and N represent the number of rows and columns of the expanded matrix F of the weight matrix after im2col algorithm. In the example, M is 8 and N is 8.

matrix is 12×16 and the vector size is artificially specified as 4, the weight expansion matrix will be divided into 12×4 vectors. Within each vector an equal proportion of the redundant weights are determined to be pruned, i.e., each vector contains the same number of zero elements. The redundant weights are not restricted in position within the vectors. This pruning pattern retains a more even distribution of weights because the number of pruned weights is the same within each vector.

The block-wise pattern [10,11] divides the weight expansion matrix into matrix blocks of size $m \times n$. For example, if the weight expansion matrix is 12×16 and the block size is artificially specified as 2×2, the weight expansion matrix will be divided into 6×8 blocks. The importance score of each block is calculated according to specific pruning criterion over the entire block.

The group-wise pattern divides the weights of different channels at the same position into a group. When the kernel tensor is expanded into a weight expansion matrix, each group forms exactly one row. At this point the unimportant rows are removed by calculating the importance score of each row according to the pruning criterion.

After removing the corresponding combination of weights from the dense model according to different pruning patterns and pruning criteria, the remaining weights form the pruned sparse neural network model.

3.3. Comparison of pruning patterns

We conduct an empirical comparison on element-wise [5], vector-wise [9], block-wise [11] and group-wise [30] pruning patterns, with different sparse ratios and tasks. The element-wise pattern uses the implementation method in [5]. The pruning criterion defines the uth index of the weight tensor W is defined as

$$score(u; W) := \frac{(W[u])^2}{\sum_{v \neq u}(W[v])^2} \tag{4}$$

As the index value increases, the score becomes smaller and smaller. After sorting each layer, the scores of all tensors are calculated and the global pruning is performed. According to the algorithm, it can be seen that this method pruning operation after the model training. The vector-wise pattern uses the implementation method in [9]. This method prunes the weights with smaller absolute values in each weight vector. The pruning schedule for this pruning method is during training. The block-wise pattern uses the method proposed in [11]. The method assigns to each block a trainable parameter m with an initial value of 1 and a range between 0 and 1. m is trained with the model and the corresponding block is pruned when this parameter is less than or equal to 0. The pruning schedule of this pruning method is also during training. The group-wise pattern uses the pattern proposed in [30]. There is no working code implementation, so the pruning criterion in DST [12] method is used to combine with the pattern. The method binds a trainable threshold at each layer and prunes groups with mean values less than the threshold. The pruning schedule for this pruning method is also during training.

Note that due to the inherent settings of pruning methods, the sparse ratios cannot be controlled to keep exactly the same. The evaluated models are VGG-16 [38], ResNet-18 [39] and Vision Transformer (ViT) [40] models. VGG-16 consists of 13 convolutional layers and 3 linear layers. ResNet-18 consists of one convolutional layer, eight residual blocks and one linear layer. Each residual block contains two convolutional layers. The ViT consists of a patch embedding layer and 12 Transformer encoders. The patch embedding layer contains a convolutional layer and each encoder contains two linear layers. We mainly perform experiments on the CIFAR-10 dataset and Tiny-Imagenet dataset, and conduct some additional experiments on the ImageNet dataset. CIFAR-10 dataset has 10 categories of images, and the corresponding task is an image classification task to predict image categories given a single image in the test set. The corresponding tasks in Tiny-ImageNet dataset and Imagenet dataset are similar to CIFAR-10. However, the number of categories in the Tiny-imagenet dataset is 200, and the number of categories in the Imagenet dataset is 1000. We also choose BERT-base model [41] as a representative in the NLP domain. It is based on the Transformer [42] implementation with 12 encoder modules. The dataset for evaluating BERT model is QQP dataset [43], which is a collection of question pairs from the community question and answer site Quora. It is a similarity and interpretation task to determine whether a pair of questions is semantically equivalent. The hardware platform we use is an Nvidia GeForce RTX 2080 Ti GPU. All pruned models are computed using cuSPARSE library.

Table 1 shows the inference time and accuracy variation (compared with non-pruned models) results of VGG-16 and ResNet-18 models with different reserved parameter ratios on the CIFAR-10 dataset. Table 2 shows the results on the Tiny-ImageNet dataset. Table 4 shows the inference time and accuracy variation of the BERT-base model on the QQP dataset.

According to the results, vector-wise and block-wise perform better than element-wise on convolutional and Transformer-based networks. However, all the pruned models have much worse performance than the corresponding dense models. Sparse computation can outperform dense computation only when the sparsity is extremely high, which will lead to significant accuracy drop and is impractical for applications. Therefore, we need a pruning pattern that can leverage an off-the-shelf dense computation library and does not need impractically high sparsity.

4. Efficient group-wise pruning

Sparse computation introduces irregular data access and is consequently time-consuming on GPUs. Through the introduce and analysis in Section 3, we can find that group-wise pruning pattern has more potential to make better use of GPU architecture. In this study, we proposes an efficient implementation of group-wise pruning pattern that enables dense computation for pruned models.

Table 1
Inference time of pruned models and dense models on CIFAR-10 dataset.

Models	Parameter (%)	Pruning pattern	Latency (ms)	Change of acc (%)
VGG-16	74.97	element-wise	6.82	+0.04
	75.00	vector-wise	6.97	+0.09
	78.74	block-wise	5.97	+0.12
	73.84	group-wise	8.63	−0.25
	49.97	element-wise	5.41	−0.23
	50.00	vector-wise	4.77	−0.32
	55.19	block-wise	5.21	−0.15
	49.90	group-wise	7.03	+0.05
	24.97	element-wise	3.27	+0.03
	25.00	vector-wise	2.55	−0.19
	25.26	block-wise	3.19	−0.38
	25.57	group-wise	2.30	−0.86
	100	dense	0.11	0.0
ResNet-18	75.99	element-wise	11.62	+0.59
	75.00	vector-wise	5.84	+0.04
	73.67	block-wise	11.93	+0.22
	77.27	group-wise	7.51	−0.35
	49.98	element-wise	10.17	+0.49
	50.00	vector-wise	3.99	−0.46
	47.41	block-wise	8.67	−0.37
	46.76	group-wise	5.86	−0.38
	24.97	element-wise	8.36	−0.06
	25.00	vector-wise	2.91	−0.53
	23.60	block-wise	5.53	−0.56
	39.35	group-wise	2.64	−1.38
	100	dense	0.70	0.0

4.1. Group-wise pattern

As introduced in Section 3.2, in the group-wise pattern the same position weights for different output channels will be clipped.

After group-wise pruning, the convolutional layer weights are expanded using the method described in Section 3.2. As shown in Fig. 4. The values of M and N are equal to $C_{in} \times h \times w$ and C_{out}, respectively. The masked parts of the figure indicate the redundant weights. It can be observed that the expanded weights to be pruned are complete rows in the matrix. By slicing and concatenating, the pruned matrix can be converted to a dense matrix. If the row vectors are removed from the weight matrix, then the corresponding column vectors in the input expansion matrix also need to be removed. Similarly, the input matrix can be converted into a dense matrix for calculation.

The above is the definition of group-wise pattern in convolutional layers of convolutional neural networks. It can also be applied to NLP models. The most heavy computation in NLP models, like RNNs and Transformer-based models, is direct multiplication of two weight matrices. We can simply follow the same idea of group-wise pruning pattern for CNNs. Each row of the weight matrix is removed or reserved simultaneously. Then the reserved rows are concatenated into a dense matrix. By this way group-wise is extended to linear layers.

4.2. Inference with group-wise pattern

Mask. When inferring with a group-wise sparse model, the model needs to know which weights have been pruned and then skips corresponding computation. Note that the pruned weights are determined by the pruning criteria. Our implementation of inference with group-wise pattern is not bound with any specific pruning criteria, so it does not suppose the exact locations of the redundant weights when we get

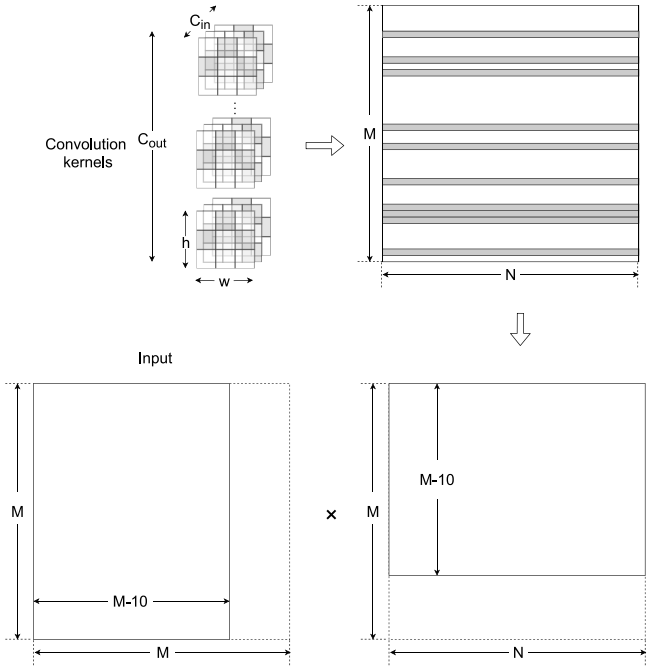


Fig. 4. Group-wise pattern. M and N represent the number of rows and columns of the expanded matrix of the weight matrix after im2col algorithm. M represent the number of rows of the expanded matrix of the input data matrix after im2col algorithm.

Algorithm 1 Inference of group-wise 2D convolutional layer

Input: input data X, layer parameters W
Output: output of this layer *output*

W_{tile} = im2col(W)
W_{groups} = W_{tile} splitted in groups
X_{tile} = im2col(X)
if use mask **then**
    $Index_{pruned}$ = [i if mask[i] is 0]
    $w = W_{groups}$
else
    $Index_{pruned}$ = [i if sum($W_{groups}[i]$) is 0]
    w = index_select($W_{groups}[i]$) if i not in $Index_{pruned}$
end if
//remove data that is not involved in the calculation
x = index_select($X_{tile}[:\ j]$) if j not in $Index_{pruned}$
$output = x \times w$
return *output*

Algorithm 2 Inference of group-wise linear layer

Input: Input data X, layer parameters W
Output: Output of this layer *output*

W_{groups} = W splitted in groups
if use mask **then**
    $Index_{pruned}$ = [i if mask[i] is 0]
    $w = W_{groups}$
else
    $Index_{pruned}$ = [i if sum($W_{tile}[i]$) is 0]
    w = index_select($W_{groups}[i]$) if i not in $Index_{pruned}$
end if
//remove data that is not involved in the calculation
x = index_select($X[:\ j]$) if j not in $Index_{pruned}$
$output = x \times w$
return *output*

Table 2
Inference time of pruned models and dense models on Tiny-ImageNet dataset.

Models	Parameters(%)	Pruning patterns	Latency(ms)	Change of acc(%)
VGG-16	72.93	element-wise	22.15	−11.52
	75.00	vector-wise	9.24	−12.34
	75.82	block-wise	17.75	−4.13
	71.74	group-wise	38.02	−4.19
	48.64	element-wise	18.54	−9.35
	50	vector-wise	8.55	−11.57
	49.2	block-wise	14.34	−4.76
	49.16	group-wise	27.86	−4.76
	24.8	element-wise	14.27	−11.71
	25	vector-wise	3.49	−11.3
	26.93	block-wise	9.89	−14.1
	30.81	group-wise	16.13	−11.68
	100	dense	0.5	0.0
ResNet-18	74.57	element-wise	47.01	−6.05
	75	vector-wise	9.59	−8.92
	75.66	block-wise	28.47	+1.2
	74.74	group-wise	30.60	+0.11
	49.27	element-wise	41.18	−4.77
	50	vector-wise	6.17	−9.87
	50.42	block-wise	18.81	−2.82
	49.23	group-wise	25.55	−0.56
	24.24	element-wise	33.87	−4.63
	25	vector-wise	3.86	−8.36
	27.66	block-wise	18.06	−4.15
	29.55	group-wise	12.98	−6.23
	100	dense	1.02	0.0

Table 3
Inference time of ViT models and dense models on CIFAR-10 and Tiny-Imagenet datasets.

Datasets	Parameters(%)	Pruning patterns	Latency(ms)	Change of acc(%)
CIFAR-10	75	element-wise	85.83	−0.56
		vector-wise	85.36	−1.08
		block-wise	28.42	+0.96
		group-wise	81.00	−1.22
	50	element-wise	57.62	−2.10
		vector-wise	58.71	−0.72
		block-wise	28.79	+0.20
		group-wise	55.55	−1.36
	25	element-wise	28.32	−0.58
		vector-wise	27.96	−0.48
		block-wise	18.19	+0.44
		group-wise	28.77	−1.95
	100	dense	3.10	0.0
Tiny-Imagenet	75	element-wise	85.98	−1.67
		vector-wise	85.54	−1.79
		block-wise	28.18	+0.37
		group-wise	81.34	0.0
	50	element-wise	57.97	−3.77
		vector-wise	58.77	−2.25
		block-wise	28.99	+0.12
		group-wise	55.99	−1.01
	25	element-wise	28.18	−3.95
		vector-wise	28.06	−2.73
		block-wise	18.05	−1.33
		group-wise	29.06	−1.12
	100	dense	3.12	0.0

a trained and pruned model. Consequently, the location information should be given when the inference starts.

Table 4
Inference time of pruned BERT-base model on QQP dataset.

Pruning pattern	Parameter (%)	Latency (ms)	Change of acc (%)
element-wise	73.46	19.66	−0.24
	43.90	20.46	−0.68
	31.66	25.32	−1.21
vector-wise	75.00	20.96	−0.19
	50.00	16.53	−0.8
	25.00	20.08	−2.42
block-wise	75.18	14.09	−0.65
	51.70	11.09	−1.64
	26.74	9.53	−2.56
group-wise	81.71	15.81	−0.28
	58.26	11.70	−1.85
	29.02	6.82	−6.32
dense	100	2.48	0.0

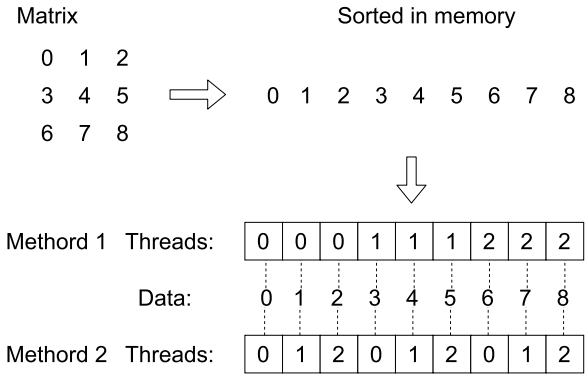


Fig. 5. Two methods of accessing data.

Existing pruning methods usually adopt two ways to keep the location of pruned weights: using masks [5], or directly setting the pruned weights to zeros [9]. If using masks, binary mask matrices are used to indicate whether the weights at corresponding locations are pruned. Our implementation covers both two ways. Algorithm 1 and Algorithm 2 show the implementations of inference with Group-wise sparse convolutional layers and linear layers, respectively. In Algorithm 1, we first split the weight W and the input data X of the current layer into W_{tile} and X_{tile} using the im2col algorithm, and divide the tiling weight into W_{groups} according to the group-wise pattern. If the model is marked with a mask, the indexes of the pruning part is extracted according to the mask. If the model is marked with changing the pruning weights to zero, the indexes of the pruning part is extracted according to the zeroing group, and the weight to be pruned is removed. According to the pruning indexes, the input data of this layer are extracted from the X_{tile} correspondingly, concatenated into a dense matrix, and then GEMM is calculated to output the calculation results of this layer. Algorithm 2 directly changes the weight of the linear layer into W_{groups} according to the group-wise pattern, and then extracts the input data with the same calculation steps as convolution to obtain the output of the linear layer. In these two algorithms, i refers to the group index of data not involved in the operation (the weight to be pruned and the input data not involved in the operation), and j refers to the group index of data to be retained.

Memory accesses coalesce. Memory accesses coalesce is the use of consecutive threads to access data at consecutive addresses. As shown in Fig. 5, a matrix of size 3×3 is accessed using 3 threads and the matrix is stored in memory in a linear fashion. There are two ways to access the matrix. The first one, thread 0 accesses the 0th, 1st, and 2nd data, and thread 1 accesses the 3rd, 4th, and 5th data, and the contiguous threads do not access contiguous memory; The second way, thread 0 accesses the 0th, 3rd, and 6th data, and thread 1 accesses the 1st, 4th,

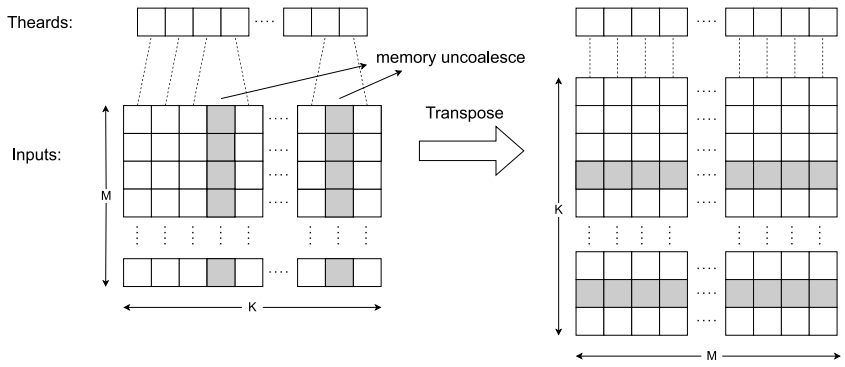


Fig. 6. Implementation of memory accesses coalesce in group-wise pattern.

and 7th data, and the contiguous threads access contiguous memory. Either way, each thread makes 3 accesses, but the second way is a coalesce memory access that requires fewer memory transactions and is therefore more efficient than the first.

In the group-wise pruning pattern, some rows of a weight matrix are pruned, so the columns of the matrix corresponding to the input data tiled at that layer then do not participate in the computation and need to be removed. When the columns of the input matrix are skipped, uncoalesced memory accesses are introduced frequently, which is inefficient on the GPU. Then the contiguous accesses to the initial input matrix become uncoalesced, which may lead to severe performance degradation. Uncoalesced memory accesses require multiple memory transactions. To alleviate this issue, the matrix can be transposed to improve its memory access efficiency, as shown in Fig. 6. In this case, column skipping becomes row skipping, eliminating uncoalesced accesses and improving access efficiency.

5. Evaluation

5.1. Setup

Benchmark. The evaluated models are VGG-16, ResNet-18, ViT and BERT-base, which cover the fields of computer vision and NLP. VGG-16 and ResNet-18 are classical CNN models. We perform inference latency evaluation on the CIFAR-10 dataset. For a convolutional layer, it is tiled after pruning. And for a linear layer, we prune it directly according to the same pattern after tiling by convolutional computation.

For the most popular family of Transformer models, we use the ViT and BERT-base models with 12-layer encoder. The ViT model is also applicable to CIFAR-10 and Tiny-Imagenet datasets for experiments. The BERT-base model downstream task being evaluated is a sentence classification task on the widely used QQP dataset.

In our experiments, the sparse CNN models and the ViT model are trained from scratch, and these models are pruned with element-wise, vector-wise, block-wise and group-wise sparse model with 100 epochs at different target sparsity levels, depending on the dataset size. The NLP models are evaluated with pre-trained models and fine-tuned by 10 epochs at each target sparsity level. They are also pruned by applying the patterns of element-wise, vector-wise, block-wise and group-wise, respectively.

Baseline. The models obtained by element-wise, vector-wise and block-wise pruning patterns are sparse models, so they are computed using the cuSPARSE library. Group-wise can be computed using the cuBLAS library through a series of processes. All experiments are performed on an NVIDIA GeForce RTX 2080 Ti GPU using FP32. The convolutional operations in the CNN models are converted to GEMM by the im2col method.

Table 5
Inference latency of group-wise pattern on CIFAR-10 dataset.

Models	Parameter (%)	Latency (ms)	Change of acc (%)
VGG-16	73.84	0.31	−0.25
	49.9	0.20	+0.05
	25.57	0.11	−0.86
ResNet-18	77.27	0.78	−0.35
	46.76	0.75	−0.38
	39.35	0.64	−1.38
ViT	75	2.80	−1.22
	50	2.37	−1.36
	25	1.96	−1.95

Table 6
Inference latency of group-wise pattern on Tiny-ImageNet dataset.

Models	Parameters(%)	Latency(ms)	Change of acc(%)
VGG-16	71.74	0.49	−4.19
	47.27	0.34	−4.76
	30.98	0.25	−11.68
ResNet-18	75.4	1.07	+0.11
	49.23	0.82	−0.56
	29.55	0.57	−6.23
ViT	75.0	2.71	−0.0
	50.0	2.37	−1.01
	25.0	1.92	−1.12

Table 7
Inference latency of group-wise pattern on QQP dataset.

Models	Parameter (%)	Latency (ms)	Change of acc (%)
BERT-base	81.71	2.43	−0.28
	58.26	2.01	−1.85
	31.21	1.47	−8.72

5.2. Result and analysis

We compare the latency of group-wise, element-wise, vector-wise and block-wise patterns on multiple models. The results of the group-wise pattern are listed in Tables 5 to 7. Figs. 7 to 10 show a comparison in inference time between the efficient group-wise introduced in this paper and the three pruning patterns element-wise, vector-wise, and block-wise in Section 3.2. The data of efficient group-wise come from Tables 5 to 7, and the data of other patterns come from Tables 1 to 4. The number of parameters on the horizontal axis is only an approximate range, rather than the exact value. For example, 25% means that with a model residual parameter of approximately 25%, it may be 27% or 23% of the true figure. The data represented in the figures are a visualization of the tables in Section 3.3 and Tables 5 to 7 in this section. It can be seen that the inference delay with group-wise pattern is significantly reduced. Supplementary experiments on Imagenet dataset are shown in Table 8.

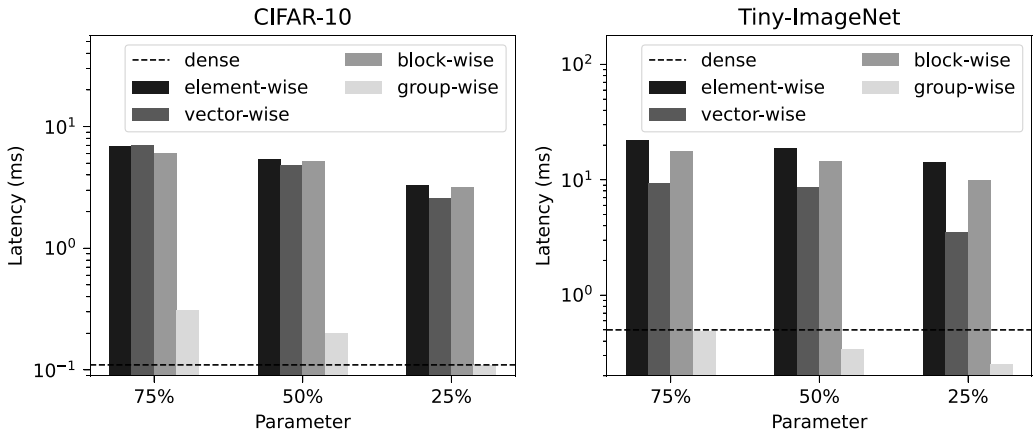


Fig. 7. This figure shows the inference latency of VGG-16 model using different pruning patterns on CIFAR-10 and Tiny-ImageNet datasets.

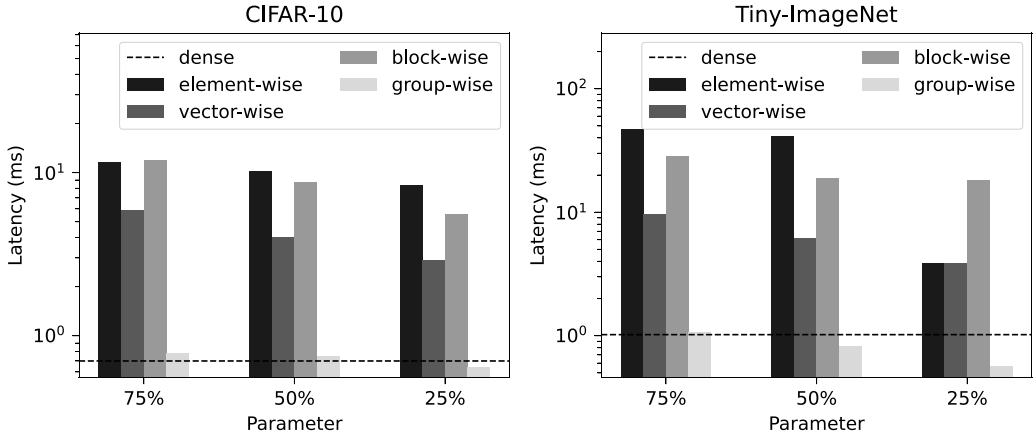


Fig. 8. This figure shows the inference latency of ResNet-18 model using different pruning patterns on CIFAR-10 and Tiny-ImageNet datasets.

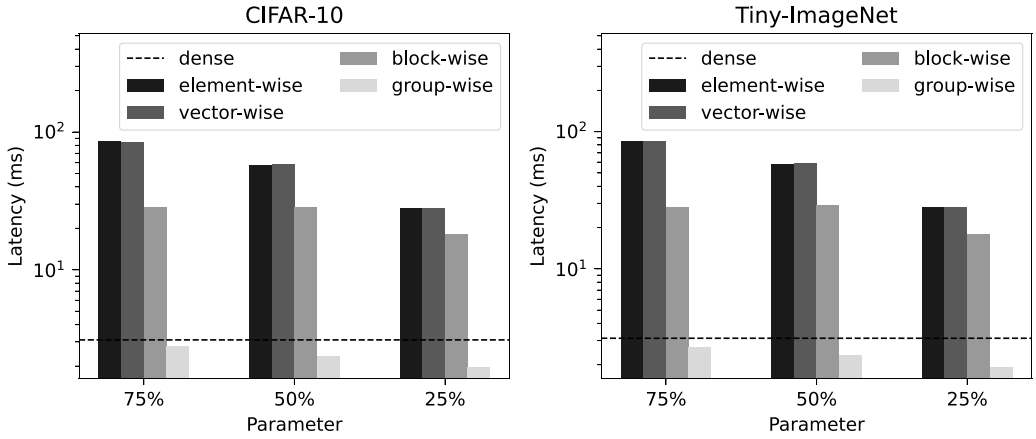


Fig. 9. This figure shows the inference latency of ViT model using different pruning patterns on CIFAR-10 and Tiny-ImageNet datasets.

The experimental results show that the group-wise pattern has shorter latency than all the other sparse patterns at the same sparsity. Group-wise effectively takes advantage of the dense GEMM acceleration, which makes fast inference possible even after pruning to obtain a sparse model. When compared with the dense model, effective latency reduction will be achieved on ResNet-18, BERT-base and ViT models. Poor performance is achieved on VGG-16 when using small datasets, but effective latency reduction is achieved on larger datasets. Because small datasets have less data to be calculated, the reduced calculation time after pruning is insufficient to counteract the overhead introduced by extra steps for pruning. Even so, when the remaining parameter ratio is 25%, the inference latency of the sparse model can be equivalent to or less than that of the dense model. And further compression can bring more acceleration. In most cases, pruned models achieve similar latency to the dense model at about 75% remaining parameter ratio, and the inference latency of the group-wise pattern will be lower than the dense model as the number of parameters continues to decrease.

Fig. 11 shows the comparison of the inference time between the group-wise implementation in this paper and Lebedev's implementation in [30]. Since [30] only focuses on convolutional layers, we take a convolutional layer from VGG-16 model as an example for comparison. The configuration of the convolutional layer set as 512 input channels,

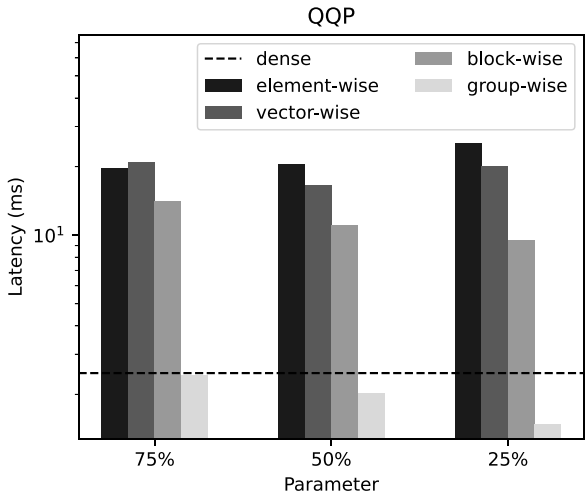


Fig. 10. This figure shows the inference latency of BERT-base model using different pruning patterns on QQP datasets.

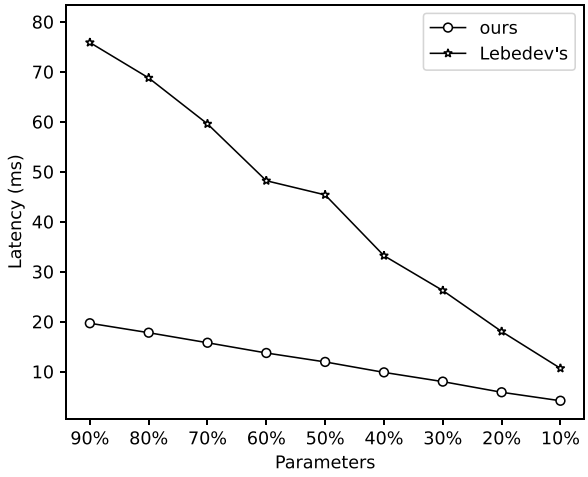


Fig. 11. Latency of different sparsity of convolutional layer using different implementations. 'Lebedev's' is the implementation in [30], and 'ours' is the implementation in this paper.

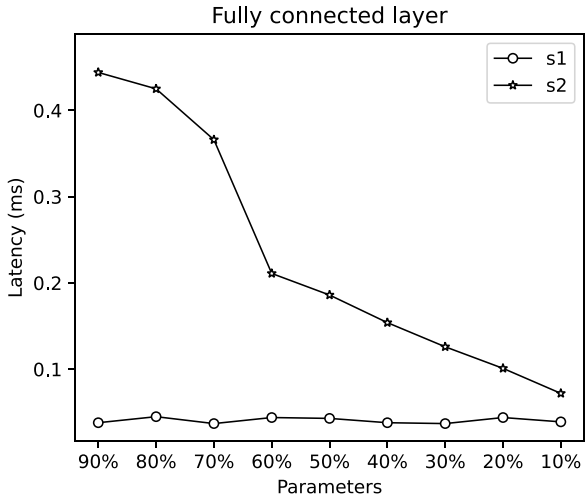


Fig. 12. Latency of group-wise pattern in linear layer. In the figure, s1, s2 correspond to the matrix concatenating and multiplication steps, respectively. The measured batch size is 64.

Table 8
When the VGG-16 model retains 50% of the parameters, use the experimental data of different patterns on full Imagenet dataset.

Parameters(%)	Pruning patterns	Latency(ms)	Change of acc(%)
50	element-wise	360.43	−5.29
	vector-wise	315.82	−8.97
	block-wise	44.25	−4.13
	group-wise	5.99	−5.26
100	dense	15.67	0.0

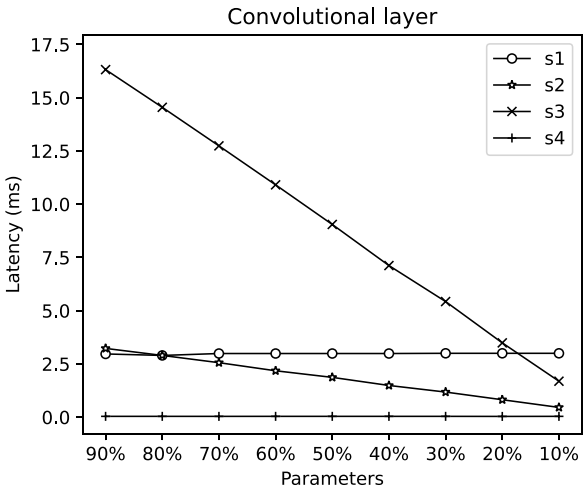


Fig. 13. Latency of group-wise pattern model in convolutional operation. In this figure, s1, s2, s3 and s4 denote im2col, matrix concatenating, multiplication, and reshaping results to feature map size steps, respectively. The measured batch size is 64.

512 output channels, 3 × 3 convolutional kernel size, 28 × 28 input data size, and 64 batch size. It can be seen that our implementation can further effectively reduce the inference latency of group-wise pruning pattern.

We also measured the latency of internal steps of convolutional layers and linear layers, as shown in Figs. 12 to 14, respectively. According to Figs. 12 to 14, the latency showed in Fig. 14 is shorter than others. The convolutional and linear layers are split into multiple kernel functions when measuring the internal steps, therefore some overhead is introduced. When measuring the time for the entire layer, it is only necessary to wait for the finish of the layer. So there is some difference in the overall time between the two sets of data.

The parameter settings for the convolutional layer are the same as the experimental settings in Fig. 11. The configuration of the linear layer: the input channel is set to 4096, the output channel is 4096, the input data size is 4096, and the batch size is 64. The above parameter settings are also from one of the layers in the VGG-16 model. Fig. 14 shows that, when the sparsity is around 20% and 10% for convolutional layer and linear layer, respectively, the inference latency is equivalent to that of dense matrix calculation. With the increase of sparsity, the latency advantage from pruning becomes more significant.

6. Conclusion

In this paper, we conduct an empirical comparison on existing mainstream pruning patterns, including element-wise, vector-wise, block-wise and group-wise pattern. After analyzing their inefficiency, we propose a more efficient implementation of the group-wise pattern on GPU using off-the-shelf GEMM library. Experimental results show that its inference latency on GPU is much lower than that of other sparse patterns. The proposed optimization implementation can further improve the inference speed of DNN models compared to existing group-wise approach. In addition, when the reserved parameters of the model are less than 75%, our group-wise inference performance can exceed that of dense models.

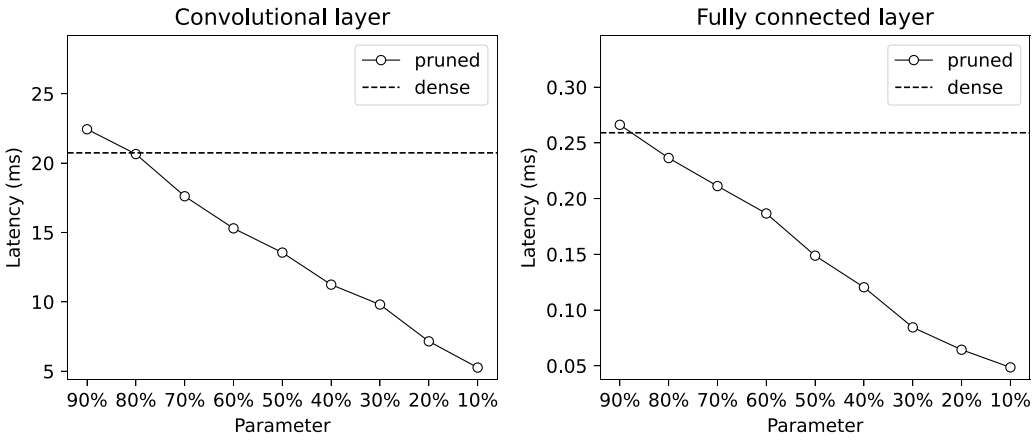


Fig. 14. Latency of different sparsity of convolutional layer and linear layer in group-wise pruned and dense model. The measured batch size is 64.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] S. Hijazi, R. Kumar, C. Rowen, et al., Using Convolutional Neural Networks for Image Recognition, Vol. 9, Cadence Design Systems Inc., San Jose, CA, USA, 2015.

[2] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, et al., Language models are unsupervised multitask learners, OpenAI Blog 1 (8) (2019) 9.

[3] G. Jain, T. Mahara, S.C. Sharma, S. Agarwal, H. Kim, TD-DNN: A time decay-based deep neural network for recommendation system, Appl. Sci. 12 (13) (2022) 6398.

[4] J. Diffenderfer, B. Kailkhura, Multi-prize lottery ticket hypothesis: Finding accurate binary neural networks by pruning a randomly weighted network, in: 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021, OpenReview.net, 2021, pp. 1–23, URL: https://openreview.net/forum?id=U_mat0b9iv.

[5] J. Lee, S. Park, S. Mo, S. Ahn, J. Shin, Layer-adaptive sparsity for the magnitude-based pruning, in: 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021, OpenReview.net, 2021, pp. 1–19, URL: https://openreview.net/forum?id=H6ATjJ0TKdf.

[6] V. Sehwag, S. Wang, P. Mittal, S. Jana, HYDRA: Pruning adversarially robust neural networks, in: H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, H. Lin (Eds.), Advances in Neural Information Processing Systems, Vol. 33, Curran Associates, Inc., 2020, pp. 19655–19666, URL: https://proceedings.neurips.cc/paper/2020/file/e3a72c791a69f87b05ea7742e04430ed-Paper.pdf.

[7] P. Hill, A. Jain, M. Hill, B. Zamirai, C. Hsu, M.A. Laurenzano, S.A. Mahlke, L. Tang, J. Mars, DeftNN: addressing bottlenecks for DNN execution on GPUs via synapse vector elimination and near-compute data fission, in: H.C. Hunter, J. Moreno, J.S. Emer, D. Sánchez (Eds.), Proceedings of the 50th Annual IEEE/ACM International Symposium on Microarchitecture, MICRO 2017, Cambridge, MA, USA, October 14-18, 2017, ACM, 2017, pp. 786–799.

[8] Z. Yao, S. Cao, W. Xiao, C. Zhang, L. Nie, Balanced sparsity for efficient dnn inference on gpu, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, 2019, pp. 5676–5683.

[9] A. Zhou, Y. Ma, J. Zhu, J. Liu, Z. Zhang, K. Yuan, W. Sun, H. Li, Learning N: M fine-grained structured sparse neural networks from scratch, in: 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021, OpenReview.net, 2021, pp. 1–15, URL: https://openreview.net/forum?id=K9bw7vqp_s.

[10] S. Narang, E. Undersander, G.F. Diamos, Block-sparse recurrent neural networks, CoRR abs/1711.02782, 2017.

[11] D.T. Vooturi, G. Varma, K. Kothapalli, Dynamic block sparse reparameterization of convolutional neural networks, in: 2019 IEEE/CVF International Conference on Computer Vision Workshops, ICCV Workshops 2019, Seoul, Korea (South), October 27-28, 2019, IEEE, 2019, pp. 3046–3053, http://dx.doi.org/10.1109/ICCVW.2019.00367.

[12] J. Liu, Z. Xu, R. Shi, R.C.C. Cheung, H.K. So, Dynamic sparse training: Find efficient sparse network from scratch with trainable masked layers, in: 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020, OpenReview.net, 2020, pp. 1–14, URL: https://openreview.net/forum?id=SJlbGJrtDB.

[13] E. Malach, G. Yehudai, S. Shalev-Shwartz, O. Shamir, Proving the lottery ticket hypothesis: Pruning is all you need, in: Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event, in: Proceedings of Machine Learning Research, vol. 119, PMLR, 2020, pp. 6682–6691.

[14] S.J. Hanson, L.Y. Pratt, Comparing biases for minimal network construction with back-propagation, in: D.S. Touretzky (Ed.), Advances in Neural Information Processing Systems 1, NIPS Conference, Denver, Colorado, USA, 1988, Morgan Kaufmann, 1988, pp. 177–185.

[15] H. Li, A. Kadav, I. Durdanovic, H. Samet, H.P. Graf, Pruning filters for efficient ConvNets, in: 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings, OpenReview.net, 2017, pp. 1–13, URL: https://openreview.net/forum?id=rJqFGTslg.

[16] P. Molchanov, S. Tyree, T. Karras, T. Aila, J. Kautz, Pruning convolutional neural networks for resource efficient inference, in: 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings, OpenReview.net, 2017, pp. 1–17, URL: https://openreview.net/forum?id=SJGCiw5gl.

[17] H. Wang, C. Qin, Y. Zhang, Y. Fu, Neural pruning via growing regularization, in: International Conference on Learning Representations, 2021, pp. 1–16.

[18] Y. LeCun, J.S. Denker, S.A. Solla, Optimal brain damage, in: D.S. Touretzky (Ed.), Advances in Neural Information Processing Systems 2, NIPS Conference, Denver, Colorado, USA, November 27-30, 1989, Morgan Kaufmann, 1989, pp. 598–605.

[19] B. Hassibi, D. Stork, Second order derivatives for network pruning: Optimal brain surgeon, Adv. Neural Inf. Process. Syst. 5 (1992).

[20] B. Hassibi, D.G. Stork, G.J. Wolff, Optimal brain surgeon and general network pruning, in: IEEE International Conference on Neural Networks, IEEE, 1993, pp. 293–299.

[21] S. Han, J. Pool, J. Tran, W. Dally, Learning both weights and connections for efficient neural network, Adv. Neural Inf. Process. Syst. 28 (2015).

[22] Y. Hu, S. Sun, J. Li, J. Zhu, X. Wang, Q. Gu, Multi-loss-aware channel pruning of deep networks, in: 2019 IEEE International Conference on Image Processing, ICIP, IEEE, 2019, pp. 889–893.

[23] G. Li, X. Ma, X. Wang, H. Yue, J. Li, L. Liu, X. Feng, J. Xue, Optimizing deep neural networks on intelligent edge accelerators via flexible-rate filter pruning, J. Syst. Archit. 124 (2022) 102431, http://dx.doi.org/10.1016/j.sysarc.2022.102431, URL: https://www.sciencedirect.com/science/article/pii/S1383762122000303.

[24] G. Li, X. Ma, X. Wang, L. Liu, J. Xue, X. Feng, Fusion-catalyzed pruning for optimizing deep learning on intelligent edge devices, IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst. 39 (11) (2020) 3614–3626, http://dx.doi.org/10.1109/TCAD.2020.3013050.

[25] X. Ma, G. Li, L. Liu, H. Liu, X. Wang, Accelerating deep neural network filter pruning with mask-aware convolutional computations on modern CPUs, Neurocomputing 505 (2022) 375–387, http://dx.doi.org/10.1016/j.neucom.2022.07.006, URL: https://www.sciencedirect.com/science/article/pii/S0925231222008669.

[26] M. Zhu, T. Zhang, Z. Gu, Y. Xie, Sparse tensor core: Algorithm and hardware co-design for vector-wise sparse neural networks on modern GPUs, in: Proceedings of the 52nd Annual IEEE/ACM International Symposium on Microarchitecture, MICRO 2019, Columbus, OH, USA, October 12-16, 2019, ACM, 2019, pp. 359–371.

[27] M. Lin, Y. Zhang, Y. Li, B. Chen, F. Chao, M. Wang, S. Li, Y. Tian, R. Ji, 1 × n pattern for pruning convolutional neural networks, IEEE Trans. Pattern Anal. Mach. Intell. (2022) 1–11, http://dx.doi.org/10.1109/TPAMI.2022.3195774.

[28] S. Anwar, K. Hwang, W. Sung, Structured pruning of deep convolutional neural networks, ACM J. Emerg. Technol. Comput. Syst. (JETC) 13 (2017) 1–18.

[29] C. Guo, B.Y. Hsueh, J. Leng, Y. Qiu, Y. Guan, Z. Wang, X. Jia, X. Li, M. Guo, Y. Zhu, Accelerating sparse DNN models without hardware-support via tile-wise sparsity, in: C. Cuicchi, I. Qualters, W.T. Kramer (Eds.), Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, SC 2020, Virtual Event / Atlanta, Georgia, USA, November 9-19, 2020, IEEE/ACM, 2020, p. 16.

[30] V. Lebedev, V.S. Lempitsky, Fast ConvNets using group-wise brain damage, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016, IEEE Computer Society, 2016, pp. 2554–2564.

[31] N. Zheng, B. Lin, Q. Zhang, L. Ma, Y. Yang, F. Yang, Y. Wang, M. Yang, L. Zhou, SparTA: Deep-learning model sparsity via Tensor-with-Sparsity-Attribute, in: 16th USENIX Symposium on Operating Systems Design and Implementation, OSDI 22, USENIX Association, Carlsbad, CA, 2022, pp. 213–232, URL: https://www.usenix.org/conference/osdi22/presentation/zheng-ningxin.

[32] K. Chellapilla, S. Puri, P. Simard, High performance convolutional neural networks for document processing, in: Tenth International Workshop on Frontiers in Handwriting Recognition, Suvisoft, 2006, pp. 1–7.

[33] Z. Liu, J. Li, Z. Shen, G. Huang, S. Yan, C. Zhang, Learning efficient convolutional networks through network slimming, in: IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017, IEEE Computer Society, 2017, pp. 2755–2763.

[34] W. Wen, C. Wu, Y. Wang, Y. Chen, H. Li, Learning structured sparsity in deep neural networks, in: D.D. Lee, M. Sugiyama, U. von Luxburg, I. Guyon, R. Garnett (Eds.), Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain, 2016, pp. 2074–2082.

[35] Y. He, G. Kang, X. Dong, Y. Fu, Y. Yang, Soft filter pruning for accelerating deep convolutional neural networks, in: J. Lang (Ed.), Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden, ijcai.org, 2018, pp. 2234–2240.

[36] P. de Jorge, A. Sanyal, H.S. Behl, P.H.S. Torr, G. Rogez, P.K. Dokania, Progressive skeletonization: Trimming more fat from a network at initialization, in: 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021, OpenReview.net, 2021, pp. 1–21, URL: https://openreview.net/forum?id=9GsFOUyUPi.

[37] N. Lee, T. Ajanthan, P.H.S. Torr, Snip: single-shot network pruning based on connection sensitivity, in: 7th International Conference on Learning Representations, ICLR 2019, New Orleans, la, USA, May 6-9, 2019, OpenReview.net, 2019, pp. 1–15, URL: https://openreview.net/forum?id=B1VZqjAcYX.

[38] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: Y. Bengio, Y. LeCun (Eds.), 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings, 2015, pp. 1–14, URL: http://arxiv.org/abs/1409.1556.

[39] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016, IEEE Computer Society, 2016, pp. 770–778.

[40] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, N. Houlsby, An image is worth 16 × 16 words: Transformers for image recognition at scale, in: 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021, OpenReview.net, 2021, pp. 1–21, URL: https://openreview.net/forum?id=YicbFdNTTy.

[41] J. Devlin, M. Chang, K. Lee, K. Toutanova, BERT: pre-training of deep bidirectional transformers for language understanding, in: J. Burstein, C. Doran, T. Solorio (Eds.), Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers), Association for Computational Linguistics, 2019, pp. 4171–4186.

[42] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA, 2017, pp. 5998–6008.

[43] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, S.R. Bowman, GLUE: A multi-task benchmark and analysis platform for natural language understanding, in: 7th International Conference on Learning Representations, ICLR 2019, New Orleans, la, USA, May 6-9, 2019, OpenReview.net, 2019, pp. 1–20, URL: https://openreview.net/forum?id=rJ4km2R5t7.

BenchCouncil Transactions on Benchmarks, Standards and Evaluations 2 (2022) 100091

Contents lists available at ScienceDirect

BenchCouncil Transactions on Benchmarks, Standards and Evaluations

journal homepage: www.keaipublishing.com/en/journals/benchcouncil-transactions-on-benchmarks-standards-and-evaluations/

Research Article

IoTBench: A data centrical and configurable IoT benchmark suite☆

Simin Chen [a,b], Chunjie Luo [a,*], Wanling Gao [a], Lei Wang [a]

[a] Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
[b] University of Chinese Academy of Sciences, Beijing, China

ARTICLE INFO

Keywords:
IoT
Benchmark
Processor
Performance evaluation
Gem5

ABSTRACT

As the Internet of Things (IoT) industry expands, the demand for microprocessors and microcontrollers used in IoT systems has increased steadily. Benchmarks provide a valuable reference for processor evaluation. Different IoT application scenarios face different data scales, dimensions, and types. However, the current popular benchmarks only evaluate the processor's performance under fixed data formats. These benchmarks cannot adapt to the fragmented scenarios faced by processors. This paper proposes a new benchmark, namely IoTBench. The IoTBench workloads cover three types of algorithms commonly used in IoT applications: matrix processing, list operation, and convolution. Moreover, IoTBench divides the data space into different evaluation subspaces according to the data scales, data types, and data dimensions. We analyze the impact of different data types, data dimensions, and data scales on processor performance and compare ARM with RISC-V and MinorCPU with O3CPU using IoTBench. We also explored the performance of processors with different architecture configurations in different evaluation subspaces and found the optimal architecture of different evaluation subspaces. The specifications, source code, and results are publicly available from https://www.benchcouncil.org/iotbench/.

1. Introduction

Internet of Things (IoT) applications are becoming more and more common, such as smart wearable devices, smart cities, smart medical care, and smart homes. With the expansion of the IoT industry, the demand for microcontrollers and microprocessors has increased steadily. Unlike general-purpose processors, which are designed for a wide range of applications, microcontrollers and microprocessors used in IoT systems are application-specific. These processors need to process different data when facing different application scenarios. For example, applications for text sequence analysis mainly deal with one-dimensional data, applications for image processing deal with two-dimensional data, and applications for video processing deal with three-dimensional data.

To realize the function and purpose of the application, it is important to choose the proper microcontroller or microprocessor. Benchmarks are useful for evaluating processors' performance. However, the current benchmarks do not pay much attention to the impact of data scale, data dimension, and data type on processors' performance, so they cannot evaluate the processor's different performances when processing different kinds of data. For example, according to [1], Dhrystone [2] consists of integer-only code, which makes it useful for micro-controllers but far from real-world applications. On the other hand, although CoreMark [3]'s data scale can be adjusted, for standard runs, the data scale (TOTAL_DATA_SIZE) must be set to 2000 bytes.

This paper proposes a new benchmark, namely IoTBench. The IoTBench workloads cover three types of algorithms commonly used in IoT applications: matrix processing, list operation, and convolution. The concept of evaluation subspace is proposed. Considering the different characteristics of the data used in different scenarios, the data space is divided into multiple evaluation subspaces according to data type, data dimension, and data scale. A set of data scales, dimensions, and types defines an evaluation subspace, and the entire data space can be divided into countless evaluation subspaces. In practice, users only need to obtain certain evaluation subspace to run the bench according to the actual scenario requirements. The three parameters of the evaluation subspace can be modified in the definition. Meanwhile, different evaluation indicators are selected to evaluate processors' performance, such as the ratio of iterations to running time (Iterations/Sec), Cycle Per Instruction (CPI), and Cache Miss Rate. We regard IoTBench as a data-centrical configurable benchmark because the main characteristic of IoTBench is that it is built to face real IoT scenarios, and the data scale, data type, and data dimension can be modified.

In the experiments, we first analyze the impact of different data types, data dimensions, and data scales on processor performance. The results show that data type, data dimension, and data scale affect the

☆ This work is supported by the Strategic Priority Research Program of the Chinese Academy of Sciences, Grant No. XDA0320000 and XDA0320300.
* Corresponding author.
E-mail addresses: chensimin22z@ict.ac.cn (S. Chen), luochunjie@ict.ac.cn (C. Luo), gaowanling@ict.ac.cn (W. Gao), wanglei_2011@ict.ac.cn (L. Wang).

https://doi.org/10.1016/j.tbench.2023.100091
Received 11 January 2023; Received in revised form 5 March 2023; Accepted 6 March 2023
Available online 8 March 2023

performance distinctly. That is to say; the data features are important factors for IoT benchmarking. We then compare ARM with RISC-V and MinorCPU with O3CPU using IoTBench. We find that the ARM ISA is more efficient than RISC-V with the same micro-architecture configuration. We explored the performance of processors with different architecture configurations in different evaluation subspaces and found the optimal architecture of different evaluation subspaces.

The contributions of this paper are as follows:

- We design and implement IoTBench, which covers three types of algorithms commonly used in IoT applications: matrix processing, list operation, and convolution. We propose the concept of evaluation subspace, which is defined by a set of data scales, dimensions, and types.
- We analyze the impact of different data types, data dimensions, and data scales on processor performance. The results show that data type, data dimension, and data scale affect the performance distinctly. We also compare ARM with RISC-V and MinorCPU with O3CPU using IoTBench. We find that the ARM ISA is more efficient than RISC-V with the same micro-architecture configuration.
- We explored the performance of processors with different architecture configurations in different evaluation subspaces and found the optimal architecture of different evaluation subspaces.

The rest of this paper is organized as follows. Section 2 presents the related works. Section 3 describes the IoTBench design and implementation. Section 4 shows the experiment settings. Section 5 provides the analysis of the experimental results. Section 6 draws the conclusions and introduces future works.

2. Related work

Benchmark is an evaluation method that has been used in the computer field for a long time. Some of the widely used benchmarks are SPEC CPU2017[4], BigDataBench [5], an industry-standardized compute-intensive benchmark, and TPC-C [6,7], a test benchmark for comparing database platforms running medium-complexity online transaction processing (OLTP) workloads. OLxPBench [8] is a composite HTAP benchmark suite. Supermarq [9] is a scalable quantum benchmark suite. TSB-UAD [10] introduces an end-to-end benchmark suite for univariate time-series anomaly detection. Galli et al. [11] provides a benchmark framework in order to analyze and discuss the most widely used and promising machine/deep learning techniques for fake news detection AIoTBench [12,13] focuses on evaluating the AI inference ability of mobile and embedded devices. MLPerf inference [14] proposes a set of rules and practices to ensure comparability across systems with wildly differing architectures. MLPerf mobile [15] is derivative of MLPerf inference which aims to Benchmark for On-Device AI. ETH Zurich AI Benchmark [16,17] aims to evaluate the AI ability of Android smartphones. These benchmarks are application-level and focus on artificial intelligence. GeekBench [18], and Moby [19] focus on benchmarking mobile phones.

Zhan [20] summarized five types of benchmark tests: measurement standard, standardized data set with defined properties, representative workload, representative data sets, and best practices, which are widely available in multiple disciplines. The benchmark proposed in this thesis is related to representative workloads and measurement standards.

There are three main benchmarks for evaluating the performance of microcontrollers and microprocessors used in IoT systems: MIPS, Dhrystone [2], and CoreMark [3]. MIPS, which is the execution of millions of instructions per second, is the most direct indicator of the processor's computational speed. However, the number and instruction types differ for different instruction set architectures. The time consumed by the execution of different instructions is also different. The value is not representative and comparable even for processors of similar architectures. Because if the instruction sequences are artificially selected, for example, selecting instruction sequences with fewer branches, the measurement results obtained will be different from the actual working and cannot accurately reflect the processor performance. Consequently, this indicator has been gradually replaced by comprehensive benchmarks such as Dhrystone.

Dhrystone is a general-purpose performance benchmark originally developed by Reinhold Weicker in 1984 with the aim of creating a short benchmarking program to measure the performance of computer system programs. Its code is composed mainly of integer operations, string operations, logical decisions, and memory accesses. Dhrystone measures processor performance by testing how many times the processor runs the Dhrystone program per second, using the VAX 11/780 as the reference machine, and reporting the results as a ratio of the number of runs on the machine to be tested to the reference machine in "DMIPS/MHz". Although Dhrystone is more meaningful than MIPS in reflecting processor performance, it is still controversial. In fact, Dhrystone's results are not only influenced by the processor's performance but also affected by factors such as the efficiency of the compiler. This characteristic allows processor manufacturers to obtain a better score by using methods such as optimizing compilers. However, this does not mean that the results of Dhrystone are meaningless. York [21] points out that when the results of Dhrystone are used for comparison, it is necessary to clearly indicate the conditions under which the benchmark is run, such as the version of Dhrystone used, the C libraries used, and so on.

CoreMark was developed by Shay Gal-On of EEMBC in 2009 to replace Dhrystone as the industry standard. CoreMark has become popular, and its features provide a strong competitive advantage. First, its code is small, easy to understand, and has good portability to ensure it runs on all platforms. Second, CoreMark introduces data that cannot be pre-computed at compile time to avoid code elimination due to compilation optimization, making all computations driven by values provided at runtime. Third, CoreMark provides rules on how to run the code and a uniform reporting format to facilitate inter-processor performance comparisons.

At present, the evaluation of IoT processors' performance is generally based on two scores, Dhrystone and CoreMark. However, the above benchmarks' standard scores only reflect the computing speed of the processor under fixed data format and have not considered processors' characteristics of data processed in different IoT applications, so they cannot meet the needs of diverse IoT scenarios. Our IoTBench considers the different data characteristics used in different scenarios; the data space is divided into multiple evaluation subspaces according to data type, dimension, and scale. Table 1 shows the comparison of IoTBench, CoreMark, and Dhrystone.

3. IoTBench

3.1. Workloads and evaluation subspace

IoTbench is comprised of list processing, matrix processing, and convolution. List processing is a kind of basic operator which is widely used in IoT scenarios. When the sensor receives the data, data cleaning and preprocessing are often performed first, and then some simple statistical analysis is carried out. In this process, search and sorting based on lists are widely used. Typical IoT scenarios, e.g., smart cities, smart homes, smartphones, and smart medical care, involve tasks such as voice control, image processing, text processing, and face recognition. Those tasks heavily depend on machine learning and deep learning. As a result, we selected the most basic operators of machine learning and deep learning, namely convolution and matrix processing.

Besides the workload itself, we argue that the data should be considered in IoT benchmarking. Different IoT scenarios face different data dimensions. For example, in scenarios that require text processing, such as natural language processing, the word vector is one-dimensional

Table 1
Comparison of IoTBench, CoreMark, and Dhrystone.

Characteristic	CoreMark	Dhrystone	IoTBench
Written in C language, portable	✓	✓	✓
Provide a single easily reportable score, concise and intuitive	✓	✓	✓
Results are independent from libraries and compilers	✓	✗	✓
Cover convolution algorithm	✗	✗	✓
Various data types can be evaluated	✗	✗	✓
Various data dimensions can be evaluated	✗	✗	✓

"✓" represents that the benchmark has this characteristic, and "✗" represents that the benchmark does not have.

Table 2
Workloads and data space.

Category	Workload	Data type	Data scale	Data dimension
List processing	List search	INT/FLOAT	Any	1/2/3
List processing	List sort	INT/FLOAT	Any	1/2/3
Matrix processing	Matrix add constant	INT/FLOAT	Any	1/2/3
Matrix processing	Matrix multiply constant	INT/FLOAT	Any	1/2/3
Matrix processing	Matrix multiply matrix	INT/FLOAT	Any	1/2/3
Convolution	Convolution	INT/FLOAT	Any	1/2/3

data. The processed image is two-dimensional data in computer vision and image processing scenarios. The processed data is three-dimensional in medical imaging, video processing, and other scenarios. Different IoT scenarios also deal with different data types. For example, in order to save computing and storage resources, AI inference on end devices often compromises between machine precision and prediction accuracy; that is, low precision, such as INT, could be used instead of high precision, such as FLOAT, for calculation. Similarly, the scale of data generated in different scenarios is different. For example, wearable devices need to monitor human body data in real time, which will generate large-scale data.

Based on the above reasons, the entire data space is divided into different evaluation subspaces according to the data scale, data dimension, and data type. A set of data scales, dimensions, and types defines an evaluation subspace, and the entire data space can be divided into countless evaluation subspaces. In practice, users only need to obtain certain evaluation subspace to run the bench according to the actual scenario requirements. The three parameters of the evaluation subspace can be modified in the macro definition. Table 2 shows the workloads and data space details.

3.2. Implementation

The data space for list processing is divided into 2 parts, list items and data items are separately stored in the 2 parts. Data structures used in list processing are shown in Fig. 1(a) and are also similar to CoreMark's. The data to be calculated together with the index is stored in structure list_data. And the structure list_data is indexed by the structure list_node, which makes up the list.

List processing consists of searching and sorting.

- List searching contains two algorithms; one is searching based on value, and the other is based on an index. IoTBench traverses the list and returns all eligible items.
- List sort is realized by merge sort and can sort the list based on value or index. Merge sort is implemented in a non-recursive way. First, every two elements in the list are divided into a group for sorting. After the group is in order, every four elements in the list are divided into a group for sorting. Expand the range of sorting to twice the present size after sorting each time until it reaches the size of the whole list.

The data structure used in convolution is shown in Fig. 1(b). Pointer 'in' points to input data, pointer 'out' points to output data, 'inWidth' refers to the width of input data, 'filter_size' refers to the kernel dimension. If the data is two-dimensional or three-dimensional, 'inHeight' is used to indicate the height of input data. If the data is three-dimensional, 'inDepth' is used to indicate the depth of input data.

One-dimensional convolution, two-dimensional convolution, and three-dimensional convolution are completed in the convolution algorithm.

- One-dimensional convolution means that the kernel slides on the vector according to the stride, and the output value is the sum of the products of the corresponding elements plus the bias.
- Two-dimensional convolution means that the kernel slides in the two-dimensional input space according to stride, and the output value is the sum of the products of the corresponding elements in the window plus the bias.
- Three-dimensional convolution means that the kernel slides in the three-dimensional input space according to stride. 3D matrix multiplication is performed in each window, and the output value is the result obtained above, plus the bias.

The data structure used in matrix processing is shown in Fig. 1(c). 'N' refers to the dimension of the matrix A/B/C. Input data is stored in matrices A and B. Output data is stored in matrix C. Matrix adds constant, matrix multiplies constant, and matrix multiplication is completed in matrix processing.

- The "matrix adds constant" function adds a constant to matrix A, and the result is stored in matrix A.
- The "matrix multiplies constant" function multiplies each item of matrix A by a constant, and the result is restored in matrix C.
- The "matrix multiplication" function multiplies matrix A and matrix B and stores the result in matrix C.

The algorithm's time complexity is shown in Table 3. $OutWidth = (InWidth - filter\_size)/stride + 1$. $OutHeight = (InHeight - filter\_size)/stride + 1$.

4. Experiment

4.1. Gem5 simulator

Gem5 simulator [22] is a modular simulation platform for computer system architecture research, including system-level architecture and processor micro-architecture, which has been widely used in academia, industry, and teaching. Gem5 was originally formed by the merger of M5[23] and GEM [24], where M5 mainly studies CPU simulation, while Gem mainly studies memory systems. Gem5 aims to create a community tool focused on architecture modeling, with flexible modeling and wide availability.

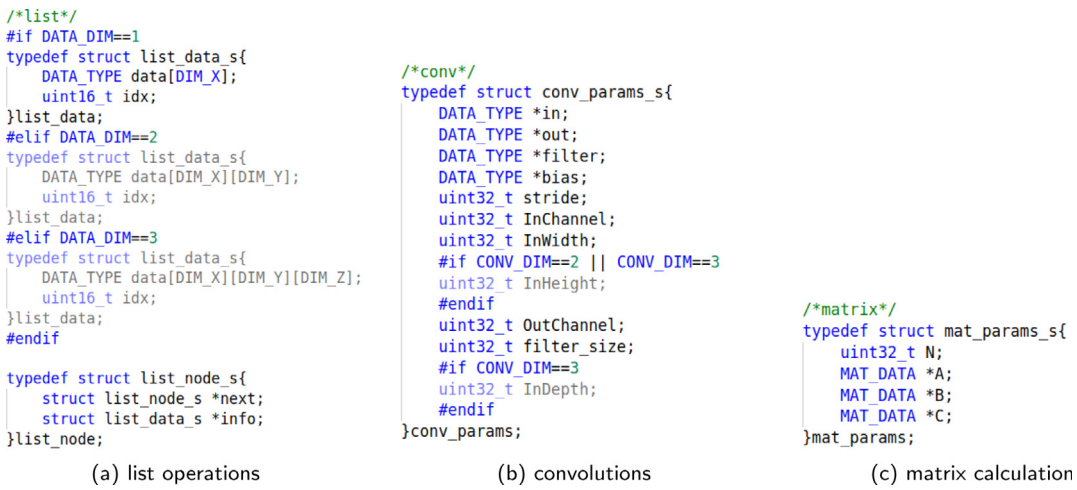


Fig. 1. Data Structure.

Table 3
Time complexity.

Algorithm	Time complexity
List search	$O(n)$
List sort	$O(n \log_2 n)$
One-dimensional convolution	$O(InChannel \cdot OutChannel \cdot filter\_size \cdot OutWidth)$
Two-dimensional convolution	$O(InChannel \cdot OutChannel \cdot filter\_size^2 \cdot OutWidth \cdot OutHeight)$
Three-dimensional convolution	$O(InChannel \cdot OutChannel \cdot filter\_size^2 \cdot OutWidth \cdot OutHeight)$
One-dimensional matrix adds/multiplies constant	$O(n)$
Two-dimensional matrix adds/multiplies constant	$O(n^2)$
Three-dimensional matrix adds/multiplies constant	$O(n^3)$
One-dimensional matrix multiplication	$O(n)$
Two-dimensional matrix multiplication	$O(n^3)$
Three-dimensional matrix multiplication	$O(n^4)$

Gem5 provides a variety of CPU models, system models, storage models, and instruction set architectures. Gem5 provides four CPU models, AtomicSimpleCPU, TimingSimpleCPU, MinorCPU (In Order), and O3CPU (Out of Order). AtomicSimpleCPU is a single IPC (that is, one clock cycle completes one instruction) CPU model that uses atomic operation to access memory. TimingSimpleCPU is similar to AtomicSimpleCPU but uses a sequential memory access model. Minor CPU is a fixed pipeline, but data structure and execution behavior can be changed. The configured in-order CPU. O3 CPU is an out-of-order CPU that is not strictly based on Alpha 21264, and unlike most simulators, Gem5 uses a model that actually executes instructions during the execution phase with high time accuracy. Gem5 also provides two system modes, system call mode (SE) and full system emulation mode (FS). The system call mode can simulate most system calls without simulating the operating system. The full system emulation mode can simulate the complete system, including the operating system, network connection, peripherals, etc. The user needs to provide the compiled Linux kernel and disk image, and the system call mode requires a longer simulation time than required. In addition, Gem5 provides two storage systems, classic mode, and ruby mode. The classic mode inherited from M5 provides a fast and easy-to-configure storage system, while the ruby mode inherited from GEM can accurately simulate storage systems that support different cache coherence protocols. At the same time, Gem5 also supports a variety of instruction set architectures, including ARM, MIPS, Power, SPARC, x86, RISC-V, etc. [22,25].

4.2. Experiment settings

We evaluate IoTBench based on Gem5 Simulator. We compare two common instruction set architectures (ISA) in IoT systems, ARM and RISC-V. In the AArch64 execution state, the A64 instruction set is used, which is a fixed-length 32-bit instruction set. We use RV64GC, an instruction set that includes compressed instructions and general-purpose instructions. We also compare in-order (Minor CPU) processors and out-of-order (O3 CPU) processors according to the way the processor executes instructions. Moreover, we evaluate various L1 and L2 Cache settings using IoTBench. The configuration of the evaluated architectures is shown in Table 4.

Table 4
Configuration of simulator.

Parameter	Value					
ISA	ARM	RISC-V				
CPU MODEL	Minor CPU	O3 CPU				
L1 ICache size	64 kB	32 kB	16 kB	8 kB	4 kB	2 kB
L1 DCache size	64 kB	32 kB	16 kB	8 kB	4 kB	2 kB
L2 Cache size	1024 kB	512 kB	0 kB			

We chose ARM and RISC-V because they are mainstream ISAs used in IoT. Also, in-order and out-of-order are two typical architectures of processors. In addition, we set the cache size according to some commercial processor manufacturers like SiFive. These settings are implemented through the command line according to the documentation of Gem5.

We use the data types INT and FLOAT in the C language; the data dimension is divided into 1 to 3 dimensions; considering that the data scale processed by the microprocessor is generally small, the data is set to two scales, namely 6144 and 12288. By modifying the DATA_SIZE, DATA_TYPE, and DATA_DIM in the macro definition, 12 evaluation subspaces are obtained. Table 5 shows the setting of the evaluation subspace in the experiment.

The cross-compilers used are aarch64-linux-gnu-gcc and riscv64-linux-gnu-gcc. ARM instruction set is Arm64, RISC-V instruction set is RV64GC; Gem5 version is 21.2.1.0. In the Gem5 directory, use SE mode to run the experiments.

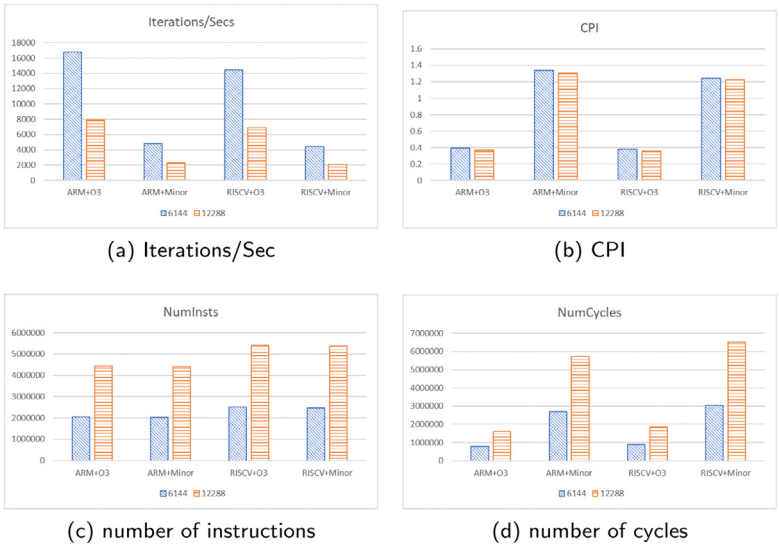


Fig. 2. Results obtained with different data scales.

Table 5
The data format of each evaluation subspace.

Evaluation subspace	DATA_SIZE/Bytes	DATA_DIM	DATA_TYPE
A	6144	1	INT_TYPE
B	6144	2	INT_TYPE
C	6144	2	FP32_TYPE
D	6144	1	FP32_TYPE
E	12288	1	INT_TYPE
F	12288	2	INT_TYPE
G	12288	2	FP32_TYPE
H	12288	1	FP32_TYPE
I	12288	3	FP32_TYPE
J	12288	3	INT_TYPE
K	6144	3	INT_TYPE
L	6144	3	FP32_TYPE

5. Results

In this section, we first analyze the impact of different data types, data dimensions, and data scales on processor performance. The results show that data type, data dimension, and data scale affect the performance distinctly. That is to say, the data features are important factors for IoT benchmarking. We then compare ARM and RISC-V with MinorCPU and O3CPU using IoTBench. We find that the ARM ISA is more efficient with the same micro-architecture configuration than RISC-V. We explore the variation of evaluation subspaces with different architecture configurations and find the different optimal architectures of different evaluation subspaces.

5.1. The impact of data feature

This subsection compares the Iterations/Sec (Iterations/Sec represents how many IoTBench iterations the processor can run per second), CPI (Cycles Per Instructions), number of instructions, and number of cycles for processors when processing data with different scale, type, and dimension, and analyzes the possible causes.

5.1.1. Data scale

As shown in Fig. 2(a), with fixed data dimensions and data types, Iterations/Sec is approximately inversely proportional to the data scale. As shown in Fig. 2(c), the number of instructions is roughly proportional to the data scale. From Fig. 2(b), CPI changes insignificantly with the data scale. It is slightly larger when the data scale is smaller.

5.1.2. Data type

As is known, with the same data size, floating-point operations are slower than integer operations. According to Fig. 3, the value of Iterations/Sec is slightly higher when the data type is int than float32. By analyzing the log of Gem5, the number of floating-point instructions accounts for less than 2% of the total instructions when the data type is float32, while integer instruction account for more than 40% and memory read/write types account for more than 50%.

5.1.3. Data dimension

As shown in Fig. 4(a), the performance is significantly better when the data dimension is one-dimensional than when the data is two-dimensional and three-dimensional. The main reason for this result is that the number of instructions is significantly lower when the data dimension is one-dimensional than when the data dimension is two-dimensional and three-dimensional (Fig. 4(c)). Through Analyzing the log of Gem5, we found that when the data is two-dimensional or three-dimensional, the integer type operations and memory read operations are about three times that of one-dimensional, and the integer multiplication operations are about six times. By analyzing the code, we found that when the data is two-dimensional or three-dimensional, it takes a lot of integer addition and multiplication operations to calculate the array index, resulting in an increase in the number of instructions. As a result, when data is one-dimensional, although the CPI is higher (Fig. 4(b)), the Iterations/Sec is still larger.

5.2. Comparison of ISAs and processor models

From Fig. 5(a), we can see that the performance of the ARM architecture is better than that of the RISC-V architecture overall with the same processor frequency; the performance of the out-of-order processor is significantly better than that of the in-order processor. From Fig. 5(c), the number of instructions of ARM architecture is less than that of RISC-V architecture. The main reason is that the ARM instruction set uses many complex instructions, such as SIMD (Single Instruction Multiple Data). The SIMD computing mode improves the computing performance but also makes the instruction set complex. And RISC-V instruction function is more simple and more basic, so the number of instructions under the RISC-V architecture will be more. Because the RISC-V architecture has more instructions, even though the CPI of the RISC-V architecture is lower than that of the ARM architecture, as shown in Fig. 5(d), the program execution cycle of the processor of the ARM architecture is still less than that of the RISC-V architecture.

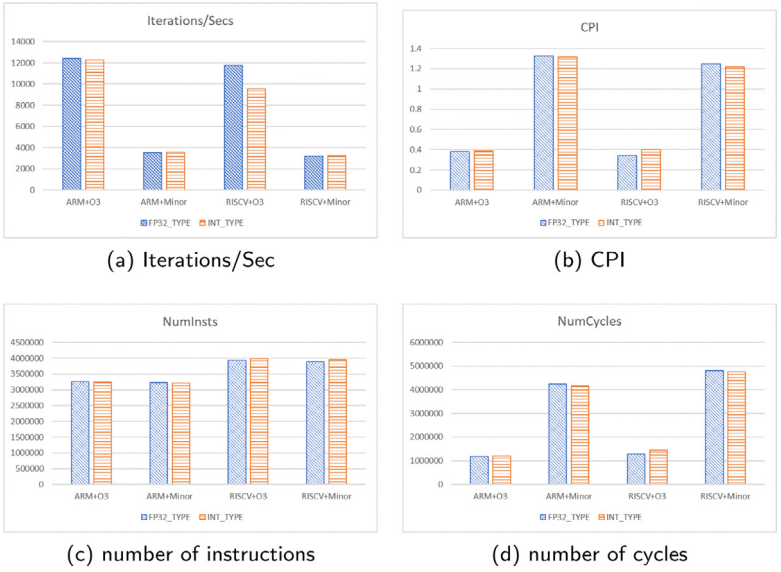


Fig. 3. Results obtained with different data types.

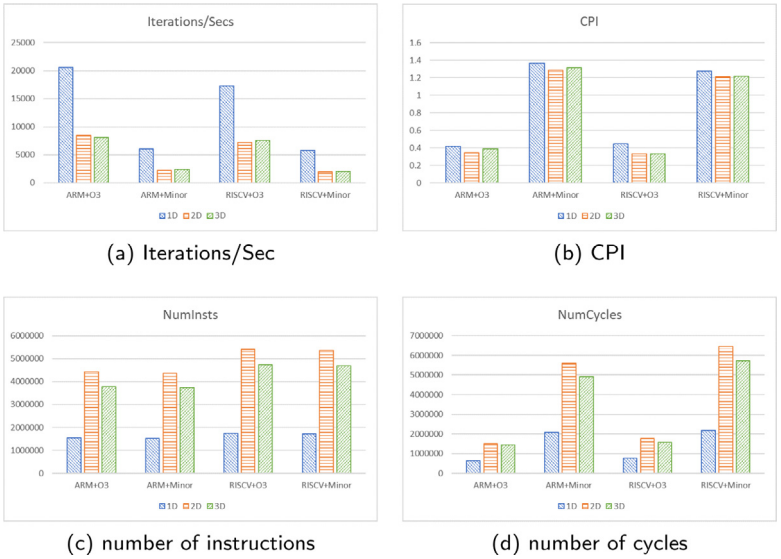


Fig. 4. Results obtained with different data dimensions.

5.3. Analysis and optimization in different evaluation subspace

This subsection analyzes the variation of processor performance with cache sizes under different CPU models and ISAs and finds the optimal cache size configuration for different subspaces.

5.3.1. Subspace A

Taking subspace A as an example, we analyze the impact of cache size on processor performance. The optimal configuration is selected based on Iterations/Sec and cache size under four configurations: ARM instruction set architecture with an out-of-order processor (ARM+O3), ARM instruction set architecture with an in-order processor (ARM+Minor), RISC-V instruction set architecture with an out-of-order processor (RISC-V+O3), and RISC-V instruction set architecture with an in-order processor (RISC-V+Minor).

When the L1 DCache is set to 16 kB, and the L1 ICache is set to 8 kB, the processor shows the same performance as both caches are set to 64 kB. L2 Cache is not set in the optimal configuration. Because the L2 Cache size is larger than the data size and the test time is short, if L2 Cache is used, the cold start will take up part of the time, making the performance drop.

Table 6 shows the configuration corresponding to the horizontal coordinate numbers in the figures below. As shown in Figs. 6(a) and 6(b), Iterations/Sec and CPI show roughly opposite trends with cache size. The increase in CPI can be attributed to the increase in miss rate due to the decrease in L1 Cache size, which causes more processor stalls. With a stable instruction count, a higher CPI implies an increase in instruction execution time, which leads to a decrease in the Iterations/Sec value. The Iterations/Sec value at numbers 5–7 increases slightly and then decreases, opposite to the L2 Cache miss rate trend. After the L1 ICache is reduced to 16 kB, the L1 ICache miss rate increases, and the number of L2 Cache access increases, resulting in a decrease in the percentage of L2 Cache cold misses and a decrease in the average miss rate. Similarly, when the L1 Cache is increased to 64 kB, the number of accesses to the L2 Cache decreases, and the average miss rate increases, causing a performance loss. Performance improves at configuration number 14 because the L2 Cache setting is eliminated here, and there is no time consumption caused by L2 Cache miss, so the performance improves. The performance drops from number 19 to

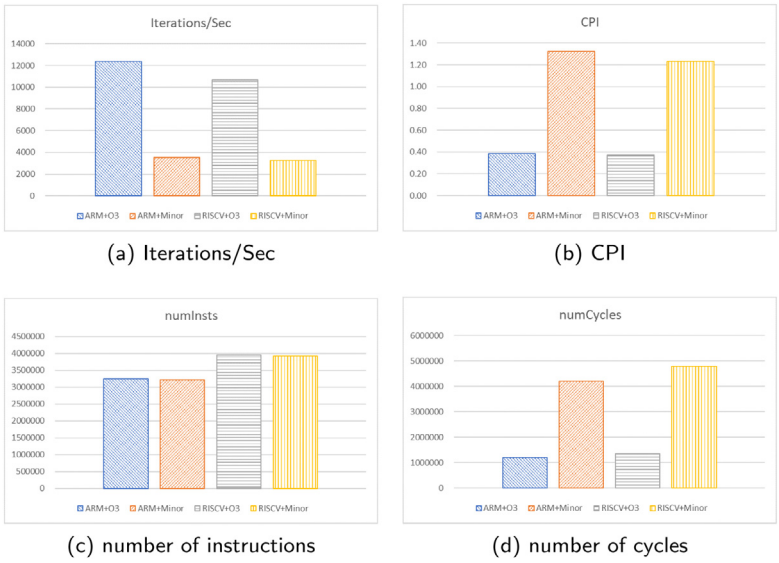


Fig. 5. Results obtained with different ISAs and CPU models.

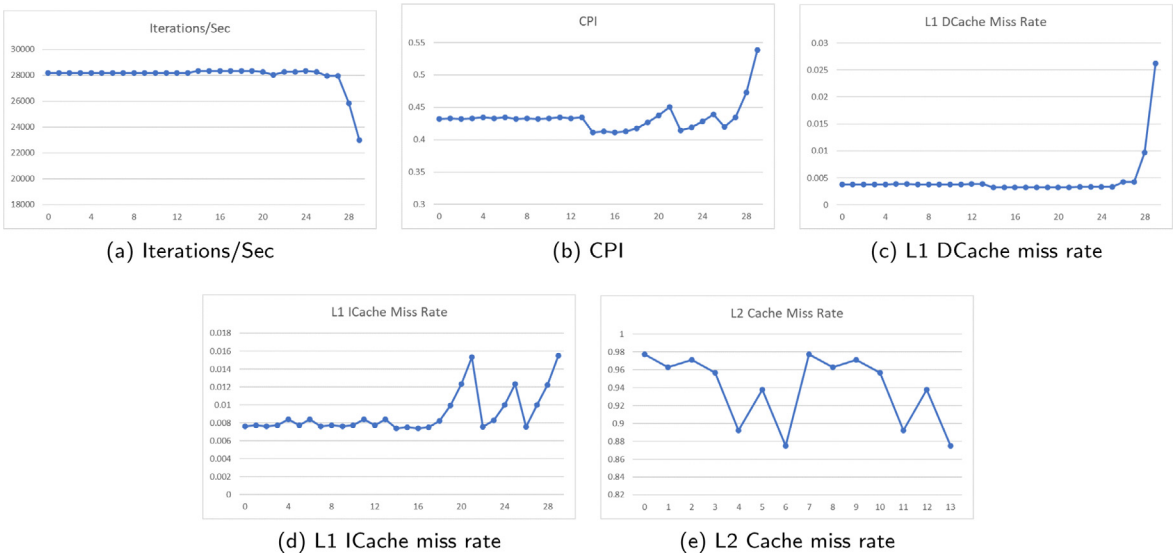


Fig. 6. Result of ARM+O3 in subspace A.

number 20 because the L1 ICache drops at this point. Combined with Fig. 6(d), we can see that the L1 ICache miss rate rises significantly here, and the same is true for numbers 23 to 25. The performance drops significantly from number 27 to number 29, when both the L1 DCache and L1 ICache drop from 8 kB to 2 kB, and both the L1 ICache miss rate and L1 DCache miss rate increase significantly.

According to Fig. 6(c), when the L1 DCache is set to 8 kB and above, its size change does not greatly affect the L1 DCache miss rate. However, below 8 kB, the miss rate varies significantly with the size of the L1 DCache because the tested data size is 6144 bytes. As shown in Fig. 6(d), when the L1 ICache is set to 16 kB and above, its size variation does not have much effect on the miss rate. However, below 16 kB, the miss rate varies significantly with size. Analyzing the first 13 sets of data with L2 Cache to get Fig. 6(e), the L2 miss rate decreases significantly at numbers 4/6/11/13, which are all configurations with L1 ICache of 16 kB. Combined with Fig. 6(d), it can be seen that the L1 ICache miss rate increases, and the number of accesses to the L2 Cache increases from the time the L1 ICache drops to 16 kB.

The performance is optimal when L1 DCache is set to 16 kB, and L1 ICache is set to 8 kB. According to Figs. 6 and 7, the results for each configuration under ARM+Minor are roughly in line with the trend of the results under ARM+O3 with cache size. However, the average value of Iterations/Sec is lower than when using the out-of-order processor, the CPI is higher, the L1 DCache miss rate and L1 ICache miss rate is significantly lower, and the L2 Cache miss rate does not change much.

When the L1 DCache is set to 16 kB and the L1 ICache is set to 8 kB, the processor achieves the best performance, the same as when both caches are set to 64 kB. According to Figs. 8 and 6, the trend of each test result with cache size under RISC-V+O3 is similar to that under ARM+O3. The average value of Iterations/Sec is lower than that of ARM architecture, CPI is higher, and the cache miss rate does not change significantly. Comparing Figs. 8(a) and 6(a) with Figs. 8(b) and 6(b), we can find that when the L1 Cache size decreases from 8 kB to 4 kB and from 4 kB to 2 kB, the performance degradation of the RISC-V group slows down, but the performance degradation of the ARM group intensifies. According to Fig. 8(c) and 6(c), the rate of increase of L1 DCache miss rate in the above interval is significantly smaller for RISC-V architecture than for ARM architecture, which may be the reason for the above phenomenon.

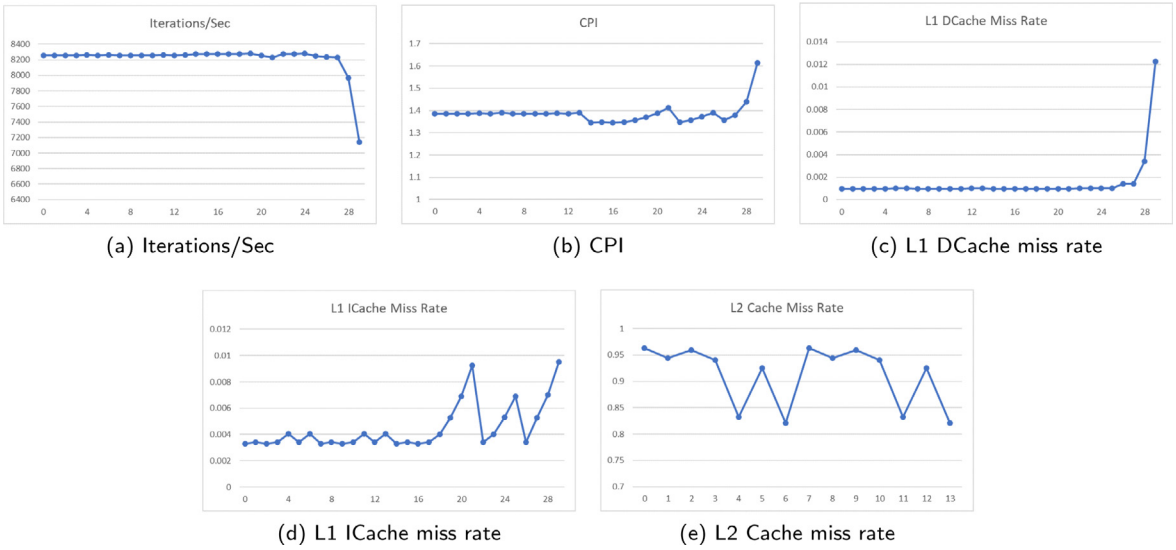


Fig. 7. Result of ARM+Minor in subspace A.

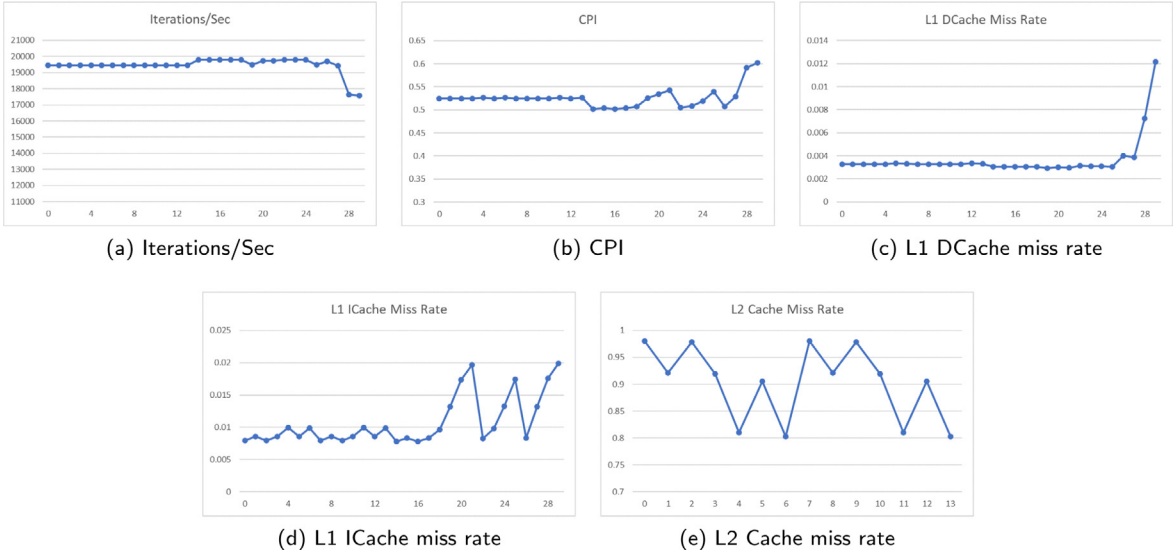


Fig. 8. Result of RISC-V+O3 in subspace A.

The performance is optimal when both caches are set to 16 kB. Comparing Figs. 9 and 8, we can see that the Iterations/Sec value decreases significantly, the CPI increases, the miss rate of both L1 DCache and L1 ICache decreases significantly, and the L2 Cache miss rate does not change significantly. Comparing Fig. 9 with Fig. 7, Iterations/Sec is relatively lower in RISC-V architecture, and CPI is also lower than the ARM architecture. The variation of performance with cache size is smaller in RISC-V architecture than in ARM architecture.

5.3.2. Other subspaces

Tables 7 summarize the optimal configurations for the 12 evaluation subspaces respectively.

The trend of processor performance with cache size varies in different evaluation subspaces, and the configuration with the best performance also varies in each evaluation subspace. Existing benchmarks such as CoreMark are tested under a fixed data size, data type, and data dimension and give a single performance score. However, IoTBench can give the final performance score under different data sizes, types, and dimensions. Users can modify the above parameters to test under what data characteristics the processor will get better performance. Users can optimize the processor for a given data space, taking into account the needs of a particular application area. It is also possible to obtain the impact of the optimization of a certain configuration on the processing of certain characteristic data. This is useful for manufacturers to produce processors for specific application areas and for users to select processors that are better suited to their data processing needs.

6. Conclusion

This paper constructs a benchmark (IoTBench) for evaluating the performance of processors in IoT scenarios. The benchmark divides the data space into multiple evaluation subspaces according to data scale, type and dimension, which aligns with IoT applications' fragmented nature. We use the Gem5 simulator to simulate processors with various configurations and use IoTBench to test the performance of each processor in different evaluation subspaces. We analyze the impact of different data types, data dimensions, and data scales on processor performance. The results show that data type, data dimension, and data scale affect the performance distinctly. The comparison shows that the ARM ISA is generally more efficient than RISC-V. The 12 evaluation

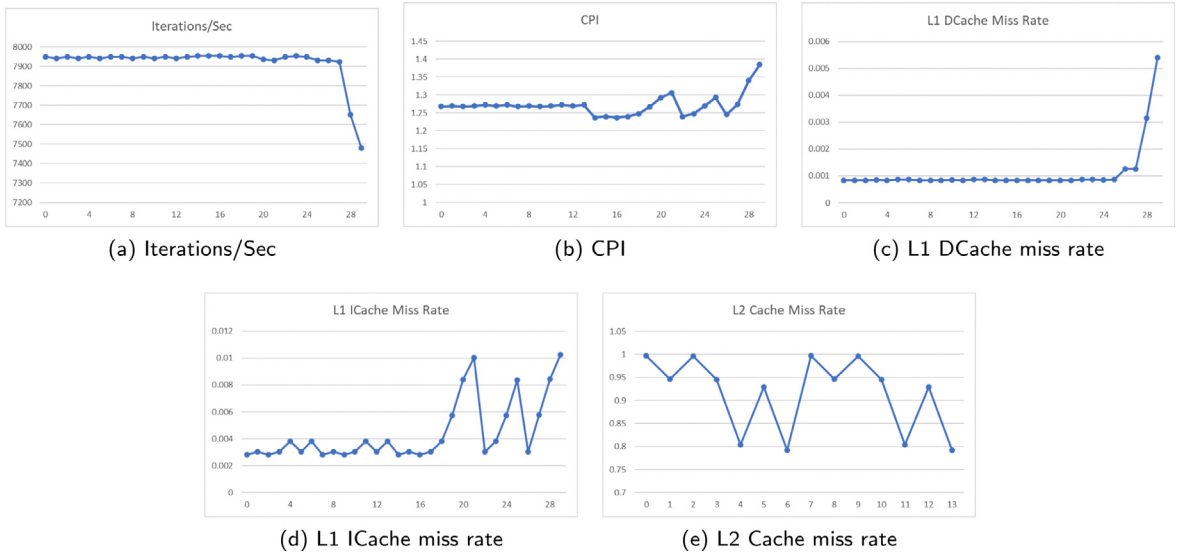


Fig. 9. Result of RISC-V+Minor in subspace A.

Table 6
The configuration corresponding to the number.

Number	L2 Cache/kB	L1 DCache/kB	L1 ICache/kB
0	1024	64	64
1	1024	64	32
2	1024	32	64
3	1024	32	32
4	1024	32	16
5	1024	16	32
6	1024	16	16
7	512	64	64
8	512	64	32
9	512	32	64
10	512	32	32
11	512	32	16
12	512	16	32
13	512	16	16
14	0	64	64
15	0	64	32
16	0	32	64
17	0	32	32
18	0	32	16
19	0	32	8
20	0	32	4
21	0	32	2
22	0	16	32
23	0	16	16
24	0	16	8
25	0	16	4
26	0	8	32
27	0	8	8
28	0	4	4
29	0	2	2

Table 7
Optimal configuration for subspace A-L.

Subspace	ISA	CPU Model	L1 DCache/kB	L1 ICache/kB	Iterations/s
A	ARM	O3	16	8	28328.61
B	ARM	O3	16	16	11695.91
C	ARM	O3	16	8	12121.21
D	ARM	O3	16	16	28571.43
E	ARM	O3	32	32	13386.88
F	ARM	O3	32	8	5173.31
G	ARM	O3	32	16	5181.35
H	ARM	O3	32	8	13458.95
I	ARM	O3	32	32	5837.71
J	ARM	O3	32	8	5621.14
K	ARM	O3	16	32	10548.52
L	ARM	O3	16	16	10964.91
A	RISC-V	O3	16	8	19801.98
B	RISC-V	O3	32	16	10718.11
C	RISC-V	O3	16	8	11013.22
D	RISC-V	O3	16	16	27247.96
E	RISC-V	O3	32	8	10070.49
F	RISC-V	O3	32	8	3915.43
G	RISC-V	O3	32	64	4738.13
H	RISC-V	O3	32	32	13192.61
I	RISC-V	O3	32	8	5208.33
J	RISC-V	O3	32	32	5130.84
K	RISC-V	O3	16	8	10152.28
L	RISC-V	O3	32	16	10214.50

subspaces obtained through IoTBench show that the same processor configuration performs differently in different evaluation subspaces, and the processor configurations corresponding to the optimal performance in different evaluation subspaces are also different. Users can set the data dimension, type, and scale of IoTBench to test different processors according to their needs to obtain processor optimization that better meets their requirements.

There are several improvements that would be made in future works. First, the experiments in this paper are conducted in the system call mode of gem5, and more experiments could be conducted in the full-system simulation mode. Second, more modules can be added to the processor configuration to test the impact of different configurations on the processors' performance. Third, More representative workloads in IoT scenarios can be selected. Fourth, the relationship between different configurations and indicators in different evaluation subspaces can be further explored.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work is supported by the Strategic Priority Research Program of the Chinese Academy of Sciences, Grant No. XDA0320000 and XDA0320300. The authors are very grateful to anonymous reviewers for their insightful feedback.

References

[1] A.R. Weiss, Dhrystone benchmark, in: History, Analysis, "Scores" and Recommendations, White Paper, ECL/LLC, 2002.

[2] R.P. Weicker, Dhrystone: a synthetic systems programming benchmark, Commun. ACM 27 (10) (1984) 1013–1030.

[3] E. Consortium, et al., Coremark, 2009.

[4] J. Bucek, K.-D. Lange, J. v. Kistowski, SPEC CPU2017: Next-generation compute benchmark, in: Companion of the 2018 ACM/SPEC International Conference on Performance Engineering, 2018, pp. 41–42.

[5] L. Wang, J. Zhan, C. Luo, Y. Zhu, Q. Yang, Y. He, W. Gao, Z. Jia, Y. Shi, S. Zhang, et al., Bigdatabench: A big data benchmark suite from internet services, in: 2014 IEEE 20th International Symposium on High Performance Computer Architecture, HPCA, IEEE, 2014, pp. 488–499.

[6] T.P.P. Council, TPC benchmark c standard specification revision 5.2, 1996, http://www.tpc.org/tpcc/spec/tpcc_current.pdf.

[7] W. Kohler, A. Shah, F. Raab, Overview of TPC benchmark c: The order-entry benchmark, in: Transaction Processing Performance Council, Technical Report, 1991.

[8] G. Kang, L. Wang, W. Gao, F. Tang, J. Zhan, OLxPBench: Real-time, semantically consistent, and domain-specific are essential in benchmarking, designing, and implementing HTAP systems, in: 2022 IEEE 38th International Conference on Data Engineering, ICDE, IEEE, 2022, pp. 1822–1834.

[9] T. Tomesh, P. Gokhale, V. Omole, G.S. Ravi, K.N. Smith, J. Viszlai, X.-C. Wu, N. Hardavellas, M.R. Martonosi, F.T. Chong, Supermarq: A scalable quantum benchmark suite, in: 2022 IEEE International Symposium on High-Performance Computer Architecture, HPCA, IEEE, 2022, pp. 587–603.

[10] J. Paparrizos, Y. Kang, P. Boniol, R.S. Tsay, T. Palpanas, M.J. Franklin, TSB-UAD: an end-to-end benchmark suite for univariate time-series anomaly detection, Proc. VLDB Endow. 15 (8) (2022) 1697–1711.

[11] A. Galli, E. Masciari, V. Moscato, G. Sperlí, A comprehensive Benchmark for fake news detection, J. Intell. Inf. Syst. 59 (1) (2022) 237–261.

[12] C. Luo, F. Zhang, C. Huang, X. Xiong, J. Chen, L. Wang, W. Gao, H. Ye, T. Wu, R. Zhou, et al., AIoT bench: towards comprehensive benchmarking mobile and embedded device intelligence, in: International Symposium on Benchmarking, Measuring and Optimization, Springer, 2018, pp. 31–35.

[13] C. Luo, X. He, J. Zhan, L. Wang, W. Gao, J. Dai, Comparison and benchmarking of ai models and frameworks on mobile devices, 2020, arXiv preprint arXiv:2005.05085.

[14] V.J. Reddi, C. Cheng, D. Kanter, P. Mattson, G. Schmuelling, C.-J. Wu, B. Anderson, M. Breughe, M. Charlebois, W. Chou, et al., Mlperf inference benchmark, in: 2020 ACM/IEEE 47th Annual International Symposium on Computer Architecture, ISCA, IEEE, 2020, pp. 446–459.

[15] V. Janapa Reddi, D. Kanter, P. Mattson, J. Duke, T. Nguyen, R. Chukka, K. Shiring, K.-S. Tan, M. Charlebois, W. Chou, et al., MLPerf mobile inference benchmark: An industry-standard open-source machine learning benchmark for on-device AI, Proc. Mach. Learn. Syst. 4 (2022) 352–369.

[16] A. Ignatov, R. Timofte, W. Chou, K. Wang, M. Wu, T. Hartley, L. Van Gool, Ai benchmark: Running deep neural networks on android smartphones, in: Proceedings of the European Conference on Computer Vision, ECCV, 2018.

[17] A. Ignatov, R. Timofte, A. Kulik, S. soo Yang, K. Wang, F. Baum, M. Wu, L. Xu, L.V. Gool, AI benchmark: All about deep learning on smartphones in 2019, in: 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), 2019, pp. 3617–3635.

[18] PrimateLabs, geekbench, https://www.geekbench.com/.

[19] Y. Huang, Z. Zha, M. Chen, L. Zhang, Moby: A mobile benchmark suite for architectural simulators, in: 2014 IEEE International Symposium on Performance Analysis of Systems and Software, ISPASS, IEEE, 2014, pp. 45–54.

[20] J. Zhan, Call for establishing benchmark science and engineering, BenchCouncil Trans. Benchmarks Stand. Eval. 1 (1) (2021) 100012, http://dx.doi.org/10.1016/j.tbench.2021.100012, URL https://www.sciencedirect.com/science/article/pii/S2772485921000120.

[21] R. York, Benchmarking in context: Dhrystone, 2002, ARM, March.

[22] N. Binkert, B. Beckmann, G. Black, S.K. Reinhardt, A. Saidi, A. Basu, J. Hestness, D.R. Hower, T. Krishna, S. Sardashti, et al., The gem5 simulator, ACM SIGARCH Comput. Archit. News 39 (2) (2011) 1–7.

[23] N.L. Binkert, R.G. Dreslinski, L.R. Hsu, K.T. Lim, A.G. Saidi, S.K. Reinhardt, The M5 simulator: Modeling networked systems, Ieee Micro 26 (4) (2006) 52–60.

[24] M.M. Martin, D.J. Sorin, B.M. Beckmann, M.R. Marty, M. Xu, A.R. Alameldeen, K.E. Moore, M.D. Hill, D.A. Wood, Multifacet's general execution-driven multiprocessor simulator (GEMS) toolset, ACM SIGARCH Comput. Archit. News 33 (4) (2005) 92–99.

[25] A. Butko, R. Garibotti, L. Ost, G. Sassatelli, Accuracy evaluation of gem5 simulator system, in: 7th International Workshop on Reconfigurable and Communication-Centric Systems-on-Chip (ReCoSoC), IEEE, 2012, pp. 1–7.

BenchCouncil Transactions on Benchmarks, Standards and Evaluations 2 (2022) 100088

Contents lists available at ScienceDirect

BenchCouncil Transactions on Benchmarks, Standards and Evaluations

journal homepage: www.keaipublishing.com/en/journals/benchcouncil-transactions-on-benchmarks-standards-and-evaluations/

Research Article

Diagnosis of COVID-19 from X-rays using combined CNN-RNN architecture with transfer learning

Md. Milon Islam [a,*], Md. Zabirul Islam [a], Amanullah Asraf [a], Mabrook S. Al-Rakhami [b], Weiping Ding [c,**], Ali Hassan Sodhro [d,e]

[a] Department of Computer Science and Engineering, Khulna University of Engineering & Technology, Khulna 9203, Bangladesh
[b] Research Chair of Pervasive and Mobile Computing, Information Systems Department, College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia
[c] School of Information Science and Technology, Nantong University, Nantong 226019, China
[d] Department of Computer Science, Kristianstad University, SE-29188 Kristianstad, Sweden
[e] Shenzhen Institutes of Advanced technology, Chinese Academy of Sciences, Shenzhen, China

ARTICLE INFO

Keywords:
COVID-19
Deep transfer learning
Chest X-rays
Recurrent neural network

ABSTRACT

Combating the COVID-19 pandemic has emerged as one of the most promising issues in global healthcare. Accurate and fast diagnosis of COVID-19 cases is required for the right medical treatment to control this pandemic. Chest radiography imaging techniques are more effective than the reverse-transcription polymerase chain reaction (RT-PCR) method in detecting coronavirus. Due to the limited availability of medical images, transfer learning is better suited to classify patterns in medical images. This paper presents a combined architecture of convolutional neural network (CNN) and recurrent neural network (RNN) to diagnose COVID-19 patients from chest X-rays. The deep transfer techniques used in this experiment are VGG19, DenseNet121, InceptionV3, and Inception-ResNetV2, where CNN is used to extract complex features from samples and classify them using RNN. In our experiments, the VGG19-RNN architecture outperformed all other networks in terms of accuracy. Finally, decision-making regions of images were visualized using gradient-weighted class activation mapping (Grad-CAM). The system achieved promising results compared to other existing systems and might be validated in the future when more samples would be available. The experiment demonstrated a good alternative method to diagnose COVID-19 for medical staff.

All the data used during the study are openly available from the Mendeley data repository at https://data.mendeley.com/datasets/mxc6vb7svm. For further research, we have made the source code publicly available at https://github.com/Asraf047/COVID19-CNN-RNN.

1. Introduction

The COVID-19 outbreak has spread rapidly due to person-to-person transmission and created a devastating impact on global health. So far, COVID-19 has infected the world with over 665,336,000 infections and close to 6,698,000 deaths [1]. Healthcare systems have been broken down in all countries due to the limited number of intensive care units (ICUs). Coronavirus-infected patients with serious conditions are admitted into ICUs. To control this pandemic, many national governments have presented 'lockdown' to ensure 'social distancing' and 'isolation' among the people to reduce person-to-person transmission [2]. The coronavirus symptoms may vary from cold to fever, acute respiratory illness, and shortage of breath [3]. The most crucial step is to diagnose COVID-19 at an early stage and isolated the infected people from others. RT-PCR is considered a key indicator to diagnose COVID-19 cases reported by the government of China [4]. However, it is a time-consuming method with a high false negatives rate [5]. In many cases, the coronavirus affected may not be identified correctly for the low sensitivity.

To combat this pandemic, a lot of interest has been found in the role of medical imaging modalities [6]. Chest radiographs such as chest X-ray and computed tomography (CT) are suitable for the detection of COVID-19 due to the high sensitivity that is already explored as a standard diagnosis system for pneumonia disease [7]. CT scan is more accurate than a chest X-ray to diagnose pneumonia but still chest X-ray is effective due to cheaper, quicker, and fewer radiation systems [8]. Deep learning [9–11] is widely used in the medical field for the analysis of complex medical images. Deep learning techniques rely on automated extracted features instead of manual handcrafted features.

* Corresponding author.
** Second corresponding author.
E-mail addresses: milonislam@cse.kuet.ac.bd (Md.M. Islam), zabir.kuet.cse@gmail.com (Md.Z. Islam), amanullahoasraf@gmail.com (A. Asraf), malrakhami@ksu.edu.sa (M.S. Al-Rakhami), dwp9988@163.com (W. Ding), ali.hassan_sodhro@hkr.se (A.H. Sodhro).

https://doi.org/10.1016/j.tbench.2023.100088
Received 11 February 2023; Received in revised form 2 March 2023; Accepted 3 March 2023
Available online 13 March 2023

We proposed a combination of CNN and RNN frameworks to identify coronavirus cases from chest X-rays in this paper. The RNN network is capable of handling long-term dependencies using internal memories. In the case of fully connected networks, nodes between layers are connectionless and process only one input but in RNN, nodes are connected from a directed graph that processes an input with a specific order [12–14]. We comparatively used four pre-trained CNN models namely VGG19, DenseNet121, InceptionV3, and Inception-ResNetV2 with RNN to find out the best CNN-RNN architecture within the limitations of the datasets. In this system, we first used the pre-trained CNN to extract significant features from images. Then, we applied the RNN classifier to identify COVID-19 cases using the extracted features. The contributions of this paper are summarized in the following.

(i) We developed and trained a combined four CNN-RNN architectures to classify coronavirus infection from others.
(ii) To detect coronavirus cases, a total of 6396 X-ray samples are used as a dataset from several sources.
(iii) An exhaustive experimental analysis is ensured to measure the performance of each architecture in terms of area under the receiver operating characteristics (ROC) curve (AUC), accuracy, precision, recall, F1-score, and confusion matrix and also applied Grad-CAM to visualize the infected region of X-rays.

The paper is organized as follows. A brief review of related works is presented in Section 2. The methods and materials including dataset collection, the development of combined networks, and performance evaluation metrics are described in Section 3. Extensive result analysis with relative discussions is illustrated in Section 4. Finally, the conclusion of the paper is drawn in Section 5.

2. Related works

Because of the COVID-19 pandemic, many efforts have been explored to develop a system for the diagnosis of COVID-19 using artificial intelligence techniques such as machine learning [15], and deep learning [16]. In this section, a detailed description is provided of the recently developed systems to diagnose COVID-19 cases.

Luz et al. [17] introduced an extended EfficientNet model based on convolutional network architecture to analyze lung conditions using X-ray images. The model used 183 samples of COVID-19 and achieved 93.9% accuracy and 80% sensitivity for coronavirus classification. Rahimzadeh and Attar [18] presented a concatenated Xception and ResNet50V2 network to find out the infected region of COVID-19 patients from chest X-rays. The network trained in eight phases and used 633 samples in each phase including 180 samples of COVID-19. The network obtained 99.56% accuracy and 80.53% recall to detect coronavirus infection. Minaee et al. [19] illustrated a deep transfer learning architecture utilizing 71 COVID-19 samples to identify infected parts from other lung diseases. The architecture obtained an overall 97.5% sensitivity and 90% specificity to differentiate coronavirus cases. Punn and Agarwal [20] demonstrated a deep neural network to identify coronavirus symptoms. The scheme used 108 COVID-19 cases and obtained an average of 97% accuracy. Khan et al. [21] introduced a deep CNN to diagnose coronavirus disease from 284 COVID-19 samples. The proposed framework found an accuracy of 89.5%, and a precision of 97% to detect coronavirus. Wang and Wong [22] presented COVID-Net to distinguish COVID-19 cases from others using chest X-ray samples. The system achieved 92.4% accuracy by analyzing 76 samples of COVID-19. Narin et al. [23] proposed deep transfer learning with three CNN architectures and used a small dataset including 50 chest X-rays for both COVID-19 and normal cases to detect coronavirus infection. The ResNet50 showed high performance with 98.6% accuracy, 96% recall, and 100% specificity among other networks.

Hemdan et al. [24] developed a COVIDX-Net framework including seven pre-trained CNN to detect coronavirus infection from X-ray samples. The dataset consisted of 25 samples of COVID-19 cases and 25 samples of normal cases. The framework obtained high performance for VGG19 with a 0.89 F1 score. Apostolopoulos and Mpesiana [25] presented a transfer learning scheme for the detection of coronavirus infection. The VGG19 obtained high performance among others with an accuracy of 93.48%, specificity of 92.85%, and sensitivity of 98.75%. Horry et al. [26] illustrated a deep transfer learning-based system and achieved the highest result for VGG19 with 83% recall and 83% precision for the diagnosis of COVID-19. Loey et al. [27] proposed a deep transfer learning approach with three pre-trained CNN networks to diagnose coronavirus disease. The dataset includes 69 COVID-19 samples, 79 pneumonia bacterial samples, 79 pneumonia virus samples, and 79 normal samples. The GoogleNet achieved an accuracy of 80.6% in the four cases scenario. Kumar and Kumari [28] used a transfer learning-based system using nine pre-trained CNNs combined with a support vector machine (SVM) to classify coronavirus-infected patients. The ResNet50-SVM showed the best performance among other models with an accuracy of 95.38%. Bukhari et al. [29] proposed a transfer learning technique for the detection of COVID-19 from X-ray samples. The system used 89 samples of COVID-19 and obtained 98.18% accuracy with a 98.19% F1-score. Abbas et al. [30] introduced a DeTraC architecture to detect coronavirus from 105 samples of COVID-19. The architecture achieved 95.12% accuracy, 91% sensitivity, 91.87% specificity, and 93.36% precision to diagnose coronavirus infection. Islam et al. [31] applied a combined CNN and LSTM architecture to classify coronavirus cases using X-ray images. The scheme applied 421 samples including 141 COVID-19 cases and achieved an accuracy of 97%, specificity of 91%, and sensitivity of 93%.

Faisal et al. [32] developed two- and three-classifier diagnosis frameworks for classifying COVID-19 patients using transfer-learning approaches that obtained an accuracy of 99.5% and 98.3% for binary and multi-class classification. Dey et al. [33] used an ML-based system with a sequence of tasks ranging from image pre-processing for the classification of COVID-19 with higher than 90% accuracy. Wang et al. [34] introduced He presented a 5G-enabled auxiliary diagnosis architecture based on federated learning for many organizations and centralized cloud cooperation to facilitate the sharing of high-generalization diagnosis tools. Singh et al. [35] illustrated a pipeline using a Hybrid Social Group Optimization algorithm to classify COVID-19 patients from chest X-rays with 99.65% accuracy. Gumaei et al. [36] developed a regression method for COVID-19 confirmed cases prediction to make future forecasting of the ongoing pandemic.

3. Methods and materials

Though some of the existing systems showed promising results, the COVID-19 dataset was quite small [19,24,27], and also the variable quality of these datasets was not addressed. It also noticed that the used dataset in those experiments was quite unbalanced which could lead to the over-classification of the majority class at the expense of the under-classification of the minority class [21,22]. On the contrary, COVID-19 images were highly inconsistent as they were retrieved from different regions of the world whereas pneumonia and normal images were uniform as well as highly curated in previous studies. Here, the COVID-19 dataset contained most adult patients, and the pneumonia dataset used mostly young patients. These discrepancies were mostly ignored in the existing systems [29]. Therefore, our proposed system used a balanced dataset with adult and young patients' images to learn the actual features of the disease. The proposed system contains several steps to diagnose COVID-19 infection as shown in Fig. 1. Firstly, in the preprocessing pipeline, chest X-ray samples were resized, shuffled, and normalized to figure out the actual features and reduce the unwanted noise from the images. Afterward, the dataset was partitioned into training and testing sets. Using the training dataset, we applied four pre-trained CNN architectures combined with the RNN classifier. The accuracy and loss of training datasets were obtained after each epoch and using a five-fold cross-validation technique, the validation loss and accuracy were found. The performance of the overall system was measured with a confusion matrix, accuracy, precision, recall, AUC, and F1-score metrics.

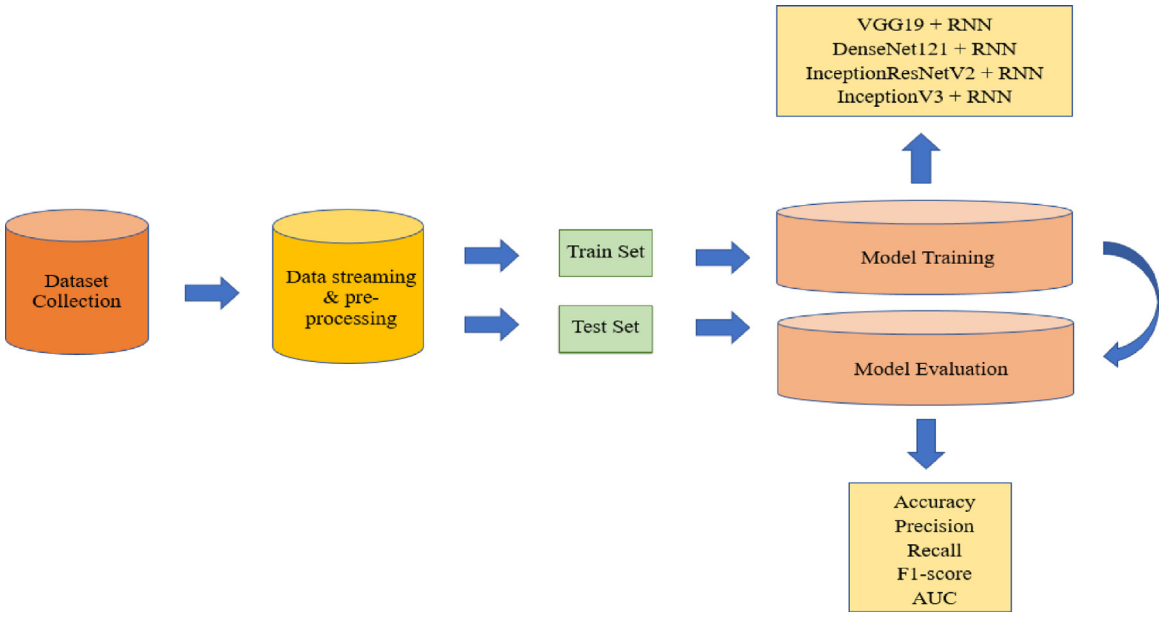


Fig. 1. The overall system architecture of the COVID-19 diagnosis framework.

Table 1
The dataset of our proposed CNN-RNN model.

Images	COVID-19	Pneumonia	Normal	Total
Training	1850	1851	1850	5551
Testing	463	462	463	1388
Total	2313	2312	2313	6939

Table 2
Characteristics of four pre-trained CNN architectures.

Network	Depth	Parameters (10^6)
VGG19	26	143.67
DenseNet121	121	8.06
InceptionV3	159	23.85
Inception-ResNetV2	572	55.87

3.1. Experimental dataset

In this paper, X-ray samples of COVID-19 were retrieved from seven different sources of the unavailability of a large specific dataset. Firstly, a total of 1401 samples of COVID-19 were collected using the GitHub repository [37,38], the Radiopaedia [39], Italian Society of Radiology (SIRM) [40], Figshare data repository websites [41,42]. Then, 912 augmented images were also collected from Mendeley instead of using data augmentation techniques explicitly [43]. Finally, 2313 samples of normal and pneumonia cases were obtained from Kaggle [44,45]. A total of 6939 samples were used in the experiment, where 2313 samples were used for each case. The total dataset was divided into 80%–20% for training and testing sets where 1850 samples of COVID-19, 1851 samples of pneumonia, and 1850 samples of normal cases were used for training including all augmented images shown in Table 1. The remaining 463 samples of COVID-19, 462 samples of pneumonia, and 463 samples of normal cases were used for the testing including only original images; no augmented images were used here. Pixel normalization was applied to images in data preprocessing step.

3.2. Development of combined network

3.2.1. Deep transfer learning with CNN

Transfer Learning [46] is an approach where information extracted by one domain is transferred to another related domain. It is applied when the dataset is not sufficient to train the parameters of any network. In this part, four pre-trained CNNs are described to accomplish the proposed CNN-RNN architecture as follows. In addition, the characteristics of four pre-trained CNN architectures are shown in Table 2.

(i) **VGG19:** VGG19 [47] is a version of the visual geometry group network (VGG) developed by Karen Simonyan and Andrew Zisserman based on deep network architecture. It has 19 layers in total including 16 convolutional layers with three fully-connected layers to perform on the ImageNet dataset [48]. VGG19 used a 3×3 convolutional filter and a stride of 1 that was followed by multiple non-linear layers. Max-pooling is applied in VGG19 to reduce the volume size of the image and achieved high accuracy in image classification.

(ii) **DenseNet121:** Dense Convolutional Network (DenseNet) [49] uses dense connections instead of direct connections among the hidden layers developed by Huang et al. In DenseNet architecture, each layer is connected to the next layer to transfer the information among the network. The feature maps are transmitted directly to all subsequent layers and use only a few parameters for training. The overfitting of a model is reduced by dense connections for small datasets. DenseNet121 has 121 layers, loaded with weights from the ImageNet dataset.

(iii) **InceptionV3:** InceptionV3 [50] is used to improve computing resources by increasing the depth and width of the network. It has 48 layers with skipped connections to use a building block and is trained on million images including 1000 categories. The inception module is repeated with max-pooling to reduce dimensionality.

(iv) **Inception-ResNetV2**: Inception-ResnetV2 [51] network is a combination of inception structure with residual connections including 164 deep layers. It has multiple-sized convolution filters trained on millions of images and avoids the degradation problem.

3.2.2. Recurrent neural network

A recurrent neural network [52] is an extended feedforward neural network with one or more feedback loops designed for processing sequential data. Given, an input sequence $(x_1, \ldots, x_t)$, an RNN generates

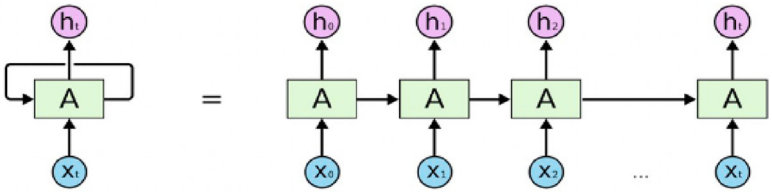


Fig. 2. The structure of recurrent neural networks.

an output sequence of $(y_1, \ldots, y_t)$ by using the following formulas, and the RNN structure is shown in Fig. 2.

$$h_t = sigm(W^{hx} X_t + W^{hh} h_{t-1}) \quad (1)$$

$$y_t = W^{yh} h_t \quad (2)$$

RNN is used whenever the input–output relationship is found based on time and capacity to handle long-term dependencies [53]. The strategy of modeling sequence is to feed the input sequence to a fixed-sized vector using an RNN, and then to map the vector to a softmax layer. Unfortunately, a problem occurs in RNN when the gradient vector is increasing and decreasing exponentially for long sequences. This vanishing gradient and exploding problem [54] create difficulties to learn long-range relationships from the sequences of the RNN architecture. However, Long Short-Term Memory (LSTM) [55] is capable to solve such a long-distance dependencies problem successfully. The main difference from RNN is that LSTM added a separate memory cell state to store long-term states and updates or exposes them whenever necessary. The LSTM consists of three gates: input gate, forget gate, and output gate where i_t denotes input gate, f_t denotes forget gate, O_t denotes output gate, $\tilde{C}$ cell input activation vector, c_t refers to the memory cell state, and h_t refers to the hidden state at each time step t.

The transition representations of LSTM are as follows.

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + V_i c_{t-1}) \quad (3)$$

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + V_f c_{t-1}) \quad (4)$$

$$O_t = \sigma(W_o x_t + U_o h_{t-1} + V_o c_{t-1}) \quad (5)$$

$$\tilde{C} = \tanh(W_c x_t + U_c h_{t-1}) \quad (6)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{C} \quad (7)$$

$$h_t = O_t \odot \tanh(c_t) \quad (8)$$

where x_t refers to current input, σ refers to the sigmoid function and $\odot$ refers to element-wise multiplication.

3.2.3. Development of CNN-RNN hybrid network

A combined method using CNN and RNN was developed for the diagnosis of COVID-19 using three types of X-ray samples in this paper. The complex features were extracted from 224 × 224 × 3 sized samples using VGG19, DeneNet121, InceptionV3, and Inception-ResNetV3. The extracted features were fed to the single-layered RNN classifier i.e. the output is produced by passing it through a single hidden state to differentiate COVID-19, pneumonia, and normal cases. The dimensionality of feature maps of pre-trained CNN and how CNN is connected to RNN were shown in Table 3.

The CNN-RNN network for COVID-19 classification is shown in Fig. 3 which contains the following steps.

Step 1: Use different pre-trained CNN models to extract essential features from X-ray images.
Step 2: Reshape the feature map into the sequence.
Step 3: Set the feature map as the input of a single-layered RNN.
Step 4: Apply a softmax classifier to classify COVID-19 X-ray images.

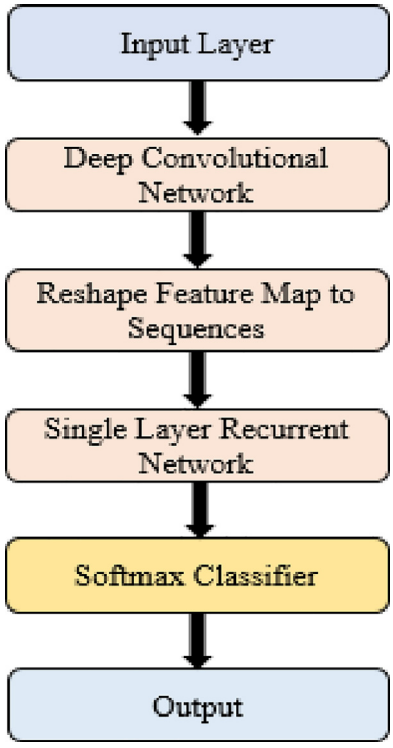


Fig. 3. The workflow of the CNN-RNN architecture for COVID-19 diagnosis.

In transfer learning, the activations of convolutional layers are the same as in original architectures. Finally, the fully connected layers were activated using Rectified Linear Unit (ReLU) [56] and the Dropout layer [57] was used in RNN layers to prevent overfitting [58] of the models. All the layers of pre-trained CNN were frozen during training except RNN and fully connected layers. Finally, the CNN-RNN architectures were trained with RMSprop [59] and a batch size of 32, a learning rate of 0.00001, and a total of 100 epochs were conducted for training. The samples were shuffled in batches between epochs. We used single-layer RNN combined with pre-trained CNN shown in Fig. 3. In a single-layer RNN, the output is produced by passing it through a single hidden state to capture the structure of a sequence.

Algorithm 1: CNN-RNN Algorithm

Input: Training data $D_{training}$, Testing data $D_{testing}$, learning rate n, epoch T, pre-trained models C, Recurrent Neural Network R, number of pre-trained models P_n
Output: Best CNN-RNN model
1. Preprocess $D_{training}$
for t = 1, 2, . . , P_n **do**
Train the model:
Obtain feature maps O: using C[t], n, and T
Reshape feature maps O into sequence H
Classify the data using R based on H
Test the model:
Evaluate performance using $D_{testing}$ and store the results and model
Compare results among the models to identify the best model
return the best CNN-RNN model

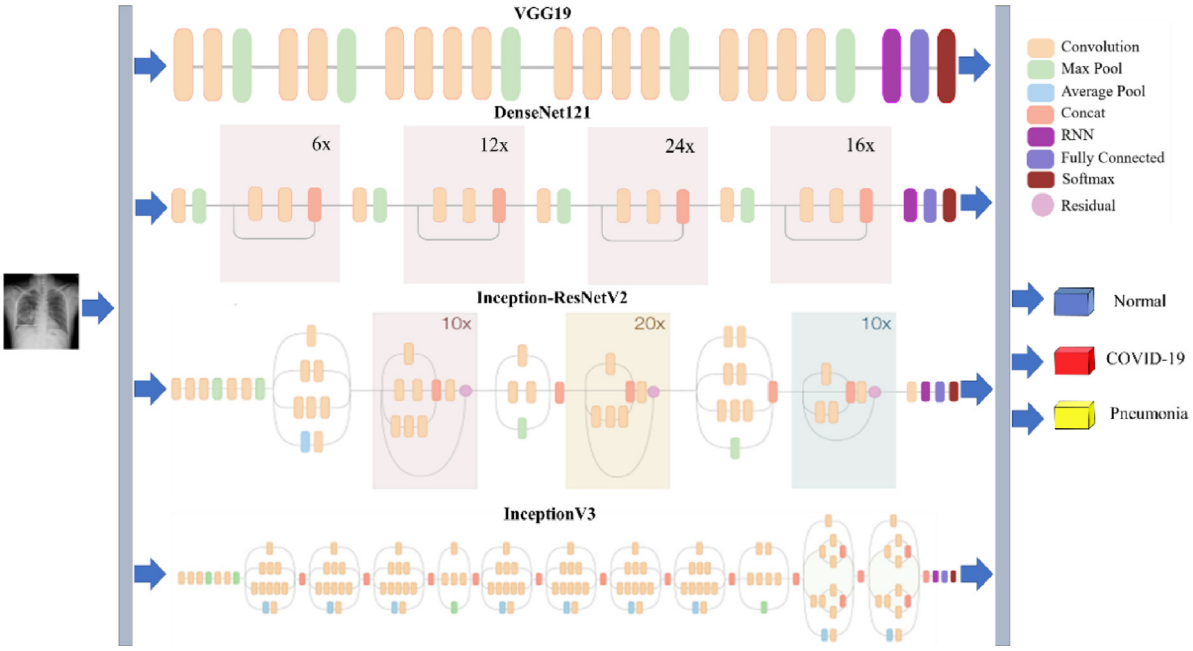


Fig. 4. The structure of the combined CNN-RNN architecture for COVID-19 diagnosis.

Table 3
The summary of the used architectures. (a) VGG19-RNN (b) DenseNet121-RNN (c) Inception-ResNetV2-RNN (d) InceptionV3-RNN.

(a)

Layer (type)	Input size	Output size
pretrained (VGG19)	224, 224, 3	7, 7, 512
reshape (Reshape)	7, 7, 512	49, 512
lstm (LSTM)	49, 512	49, 512
flatten (Flatten)	49, 512	25088
fc_1 (Dense)	25088	4096
fc_2 (Dense)	4096	4096
output (Dense)	4096	3

(b)

Layer (type)	Input size	Output size
pretrained (DenseNet121)	224, 224, 3	7, 7, 1024
reshape (Reshape)	7, 7, 1024	49, 1024
lstm (LSTM)	49, 1024	49, 1024
flatten (Flatten)	49, 1024	50176
fc_1 (Dense)	50176	4096
fc_2 (Dense)	4096	4096
output (Dense)	4096	3

(c)

Layer (type)	Input size	Output size
pretrained (Inception-ResNetV2)	224, 224, 3	5, 5, 1536
reshape (Reshape)	5, 5, 1536	25, 1536
lstm (LSTM)	25, 1536	25, 512
flatten (Flatten)	25, 512	12800
fc_1 (Dense)	12800	4096
fc_2 (Dense)	4096	4096
output (Dense)	4096	3

(d)

Layer (type)	Input size	Output size
pretrained (InceptionV3)	224, 224, 3	5, 5, 2048
reshape (Reshape)	5, 5, 2048	25, 2048
lstm (LSTM)	25, 2048	25, 512
flatten (Flatten)	25, 512	12800
fc_1 (Dense)	12800	4096
fc_2 (Dense)	4096	4096
output (Dense)	4096	3

We have used RNN as a sequence-to-sequence layer and taken the output sequence as input for fully-connected layers downstream in the developed system. In this architecture, Gradient clipping is used to handle the long sequence problem. In our proposed system, the order of elements of a sequence is horizontal. Basically, to develop the sequence by collecting the pixels from the images in three orders such as horizontal, vertical and spiral are used. The structure of the combined CNN-RNN is shown in Fig. 4. Algorithm 1 presented the proposed CNN-RNN technique to detect COVID-19 cases.

3.3. Evaluation criteria

The performance of the developed system is measured in terms of AUC, accuracy, precision, recall, and F1 score. The evaluation metric parameters are represented mathematically in the following. Here, correctly classified COVID-19 cases are denoted by True Positive (TP), correctly classified pneumonia or normal cases are represented by True Negative (TN), wrongly classified as COVID-19 cases are denoted by False Positive (FP), and wrongly classified as pneumonia or normal cases are depicted by False Negative (FN).

$$\text{Accuracy} = (TP + TN)/(TN + FP + TP + FN) \tag{9}$$

$$\text{Precision} = TP/(TP + FP) \tag{10}$$

$$\text{Recall} = TP/(TP + FN) \tag{11}$$

$$F1 - \text{score} = (2 * \text{Precision} * \text{Recall})/(\text{Precision} + \text{Recall}) \tag{12}$$

4. Results analysis

All the experiments were performed on a Google Colaboratory Linux server with Ubuntu 16.04 operating system using a Tesla K80 GPU graphics card and the TensorFlow/Keras framework of python language.

4.1. Results analysis

The accuracy and loss curves in the training and validation phases are shown in Fig. 5. For VGG19-RNN architecture, the highest training and validation accuracy is observed at 99.01% and 97.74% and loss is

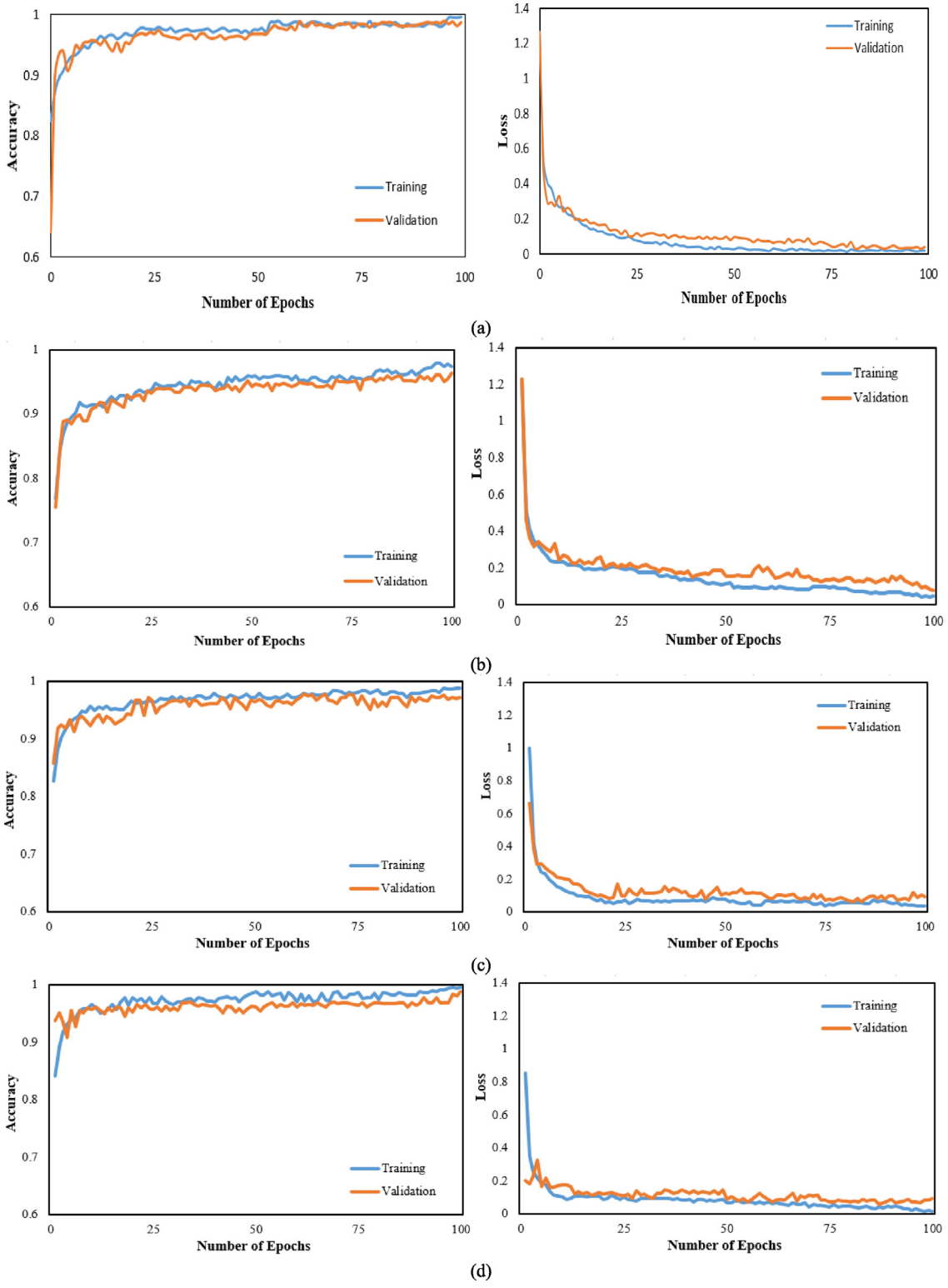


Fig. 5. Accuracy and loss curve of four CNN-RNN architectures. (a) VGG19 (b) DenseNet121(c) InceptionV3 (d) Inception-ResNetV2.

0.02 and 0.09 at epoch 100. On the contrary, the lowest training and validation accuracy is obtained 98.03% and 94.91% and loss is 0.05 and 0.26 at epoch 100 for the InceptionV3-RNN network. Analyzing the loss curve, it is seen that the loss values of VGG19-RNN decrease faster and tends to zero than other networks.

Fig. 6 demonstrates the confusion matrix of the developed architectures. Among 1388 samples, 2 samples were misclassified by the VGG19-RNN network including only one sample for COVID-19 cases, 3 samples were misclassified by the DenseNet121-RNN network including two COVID-19 samples, 20 samples were misclassified by InceptionV3-RNN architecture consisting of three COVID-19 samples and 7 samples were misclassified by the Inception-ResNetV2-RNN network comprising of seven COVID-19 samples. Hence, it was found that VGG19-RNN architecture is superior to other networks and selected as

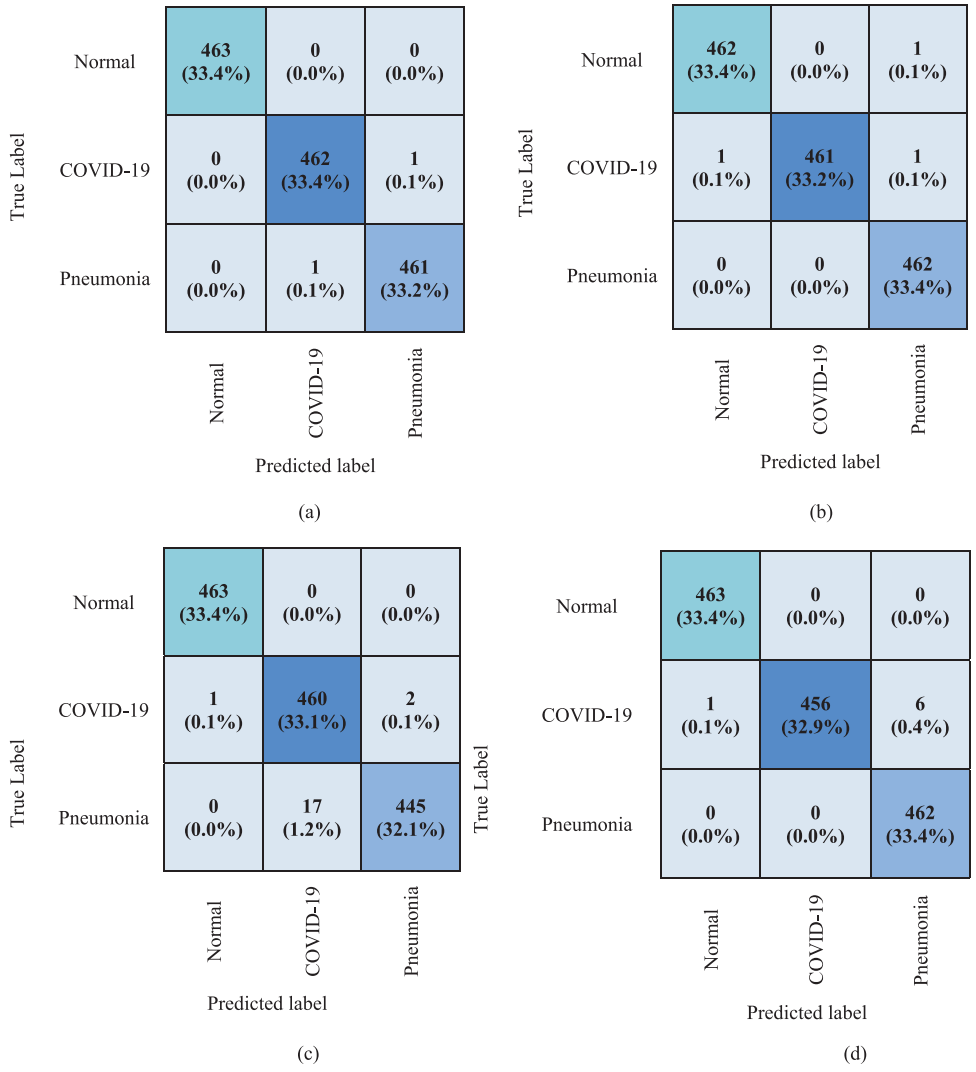


Fig. 6. Confusion matrix of the CNN-RNN architecture for COVID-19 diagnosis. (a) VGG19 (b) DenseNet121 (c) InceptionV3 (d) Inception-ResNetV2.

a main deep learning architecture with high performance. Moreover, Table 4 illustrates a comparison between the CNN-RNN network used in these experiments in terms of computational times. It is observed that VGG19-RNN achieved the highest performance and took 16722.41s for training and 129.69s for testing. In addition, it is also noticed that InceptionV3-RNN needed 16376.09s for training and 170.14s for testing.

Though InceptionV3-RNN model required less training time and more testing time than the VGG19-RNN model. It is concluded that the researcher has the choice to select the deep learning model between accuracy and computational time to use, but in the medical field, accuracy is always the main criterion. Hence, the experimental result revealed that the VGG19-RNN model outperforms other CNN-RNN architectures.

In this paper, the performance of four CNN-RNN architectures is summarized in Table 5. The best performance was found by the VGG19-RNN network with 99.86% accuracy, 99.99% AUC, 99.78% precision, 99.78% recall, and 99.78% F1-score for COVID-19 cases. On the contrary, the comparatively low performance was obtained by InceptionV3-RNN architecture with 98.56% accuracy, 99.95% AUC, 99.35% precision, 96.44% recall, and 97.87% F1-score. Besides, ROC curves were also added between TP and FP rates for all networks shown in Fig. 7. The networks can differentiate COVID-19 cases from others with an AUC in the range of 99.95% to 99.99%. For better visualization and to show the differences between the classifiers Precision–Recall (PR) curve is also added shown in Fig. 8.

Table 4
Comparative computational time CNN-RNN models.

Model	Training time (s)	Testing time (s)
VGG19	16722.41	129.69
DenseNet121	18145.67	196.02
InceptionV3	16376.09	170.14
Inception-ResnetV2	17727.26	310.63

Finally, Grad-CAM is applied which refers to a heat map to highlight class-specific regions of chest X-rays. Fig. 9 shows the heatmaps and superimposed images of COVID-19, pneumonia, and normal cases for the VGG19-RNN network.

4.2. Discussions

In this paper, the combination of four CNNs and RNNs was used to diagnose the COVID-19 infection. The results demonstrated that VGG19-RNN is more effective to differentiate COVID-19 cases from

Table 5
Performance of the combined CNN-RNN architecture.

Classifier	Patient status	AUC (%)	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
VGG19-RNN	COVID-19	99.99	99.86	99.78	99.78	99.78
	Pneumonia		99.86	99.78	99.78	99.78
	Normal		99.86	100.0	100.0	100.0
DenseNet121-RNN	COVID-19	99.99	99.78	99.57	100.0	99.78
	Pneumonia		99.78	100.0	99.57	99.78
	Normal		99.78	99.78	99.78	99.78
InceptionV3-RNN	COVID-19	99.95	98.56	99.35	96.44	97.87
	Pneumonia		98.56	96.32	99.55	97.91
	Normal		98.56	100.0	99.78	99.89
Inception-ResNetV2- RNN	COVID-19	99.99	99.50	98.49	100.0	99.24
	Pneumonia		99.50	100.0	99.72	99.86
	Normal		99.50	100.0	99.78	99.89

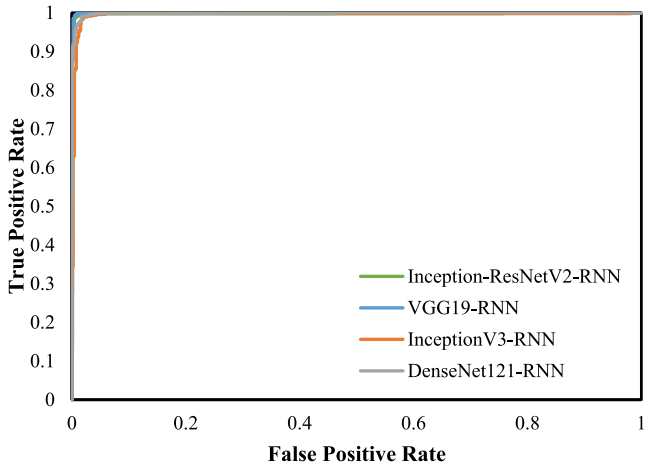


Fig. 7. ROC curve of four combined CNN-RNN networks.

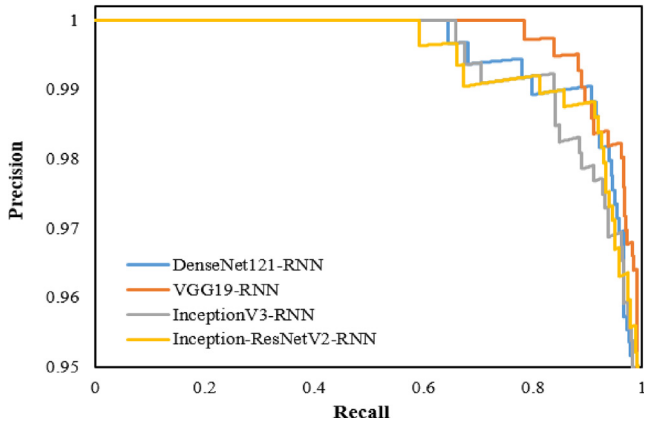


Fig. 8. PR curve of four combined CNN-RNN networks.

pneumonia and normal cases and is considered as a main deep learning architecture. A comparison between simple CNN-based pre-trained networks with our study is demonstrated in Table 6. It is clearly shown that the VGG19-RNN network has obtained higher performance than pre-trained CNN networks. Finally, another comparison between recent works with our study is demonstrated in Table 7. It is observed that existing systems can distinguish coronavirus infection with accuracy in the range of 80.6% to 99.6%. On the contrary, the VGG19-RNN network obtained 99.9% accuracy which is higher than other existing

Table 6
Comparison between pre-trained CNN with CNN-RNN architecture based on COVID-19 patients.

Classifier	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
VGG19	99.63	99.51	99.65	99.58
DenseNet121	99.26	99.45	99.38	99.41
InceptionV3	99.13	98.71	98.92	98.81
Inception-ResNetV2	99.28	99.38	98.06	98.72
VGG19-RNN	99.86	99.78	99.78	99.78

systems. In addition, a comparison in terms of computational time showed that [24] took 2641.0s for training 40 images and 4.0s for testing 10 images, [60] consumed 2277.6s for training 8997 images, [61] required 79184.3s and 262.0s for training and testing 4449 and 1638 images respectively. In our experiment, VGG19-RNN architecture took 16722.4s and 129.7s for training and testing 5551 and 1388 images respectively which is comparatively faster than other existing models. Hence, finally, it is evident that the VGG19-RNN network showed good performance compared to other studies.

5. Conclusion

During the COVID-19 pandemic, the use of deep learning techniques for the diagnosis of COVID-19 has become a crucial issue to overcome the limitation of medical resources. In this work, we used CNN with deep transfer learning and RNN to classify the X-ray samples into three categories: pneumonia, COVID-19, and normal. The four popular CNN networks were used to extract features, which were then applied by the RNN network to identify different classes. The VGG19-RNN is considered the best network with 99.9% accuracy, 99.9% AUC, 99.8% recall, and 99.8% F1-score to detect COVID-19 cases. Hopefully, it would reduce the workload for the doctor to test COVID-19 cases.

There are some limitations to our proposed system. First, the COVID-19 samples are small that need to be updated with more samples to validate our proposed system. Second, this experiment only works with a posterior–anterior view of chest X-ray, hence it is not able to effectively classify other views such as apical, lordotic, etc. Third, the performance of our experiment is not compared with radiologists which would be our future work.

Funding statement

None

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

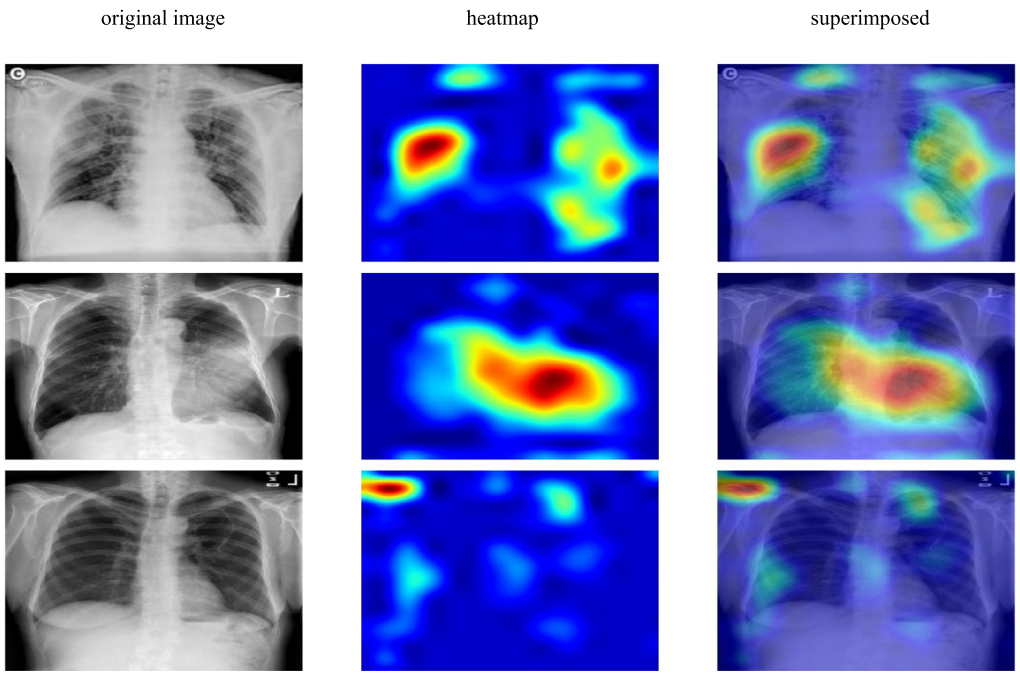


Fig. 9. First, second and third rows represent the samples of COVID-19, pneumonia, and normal correspondingly. Besides, the first, second, and third columns refer to the original, heatmap and superimposed images for VGG19-RNN.

Table 7
Comparative study of the proposed CNN-RNN architecture with existing works concerning accuracy.

Author	Architecture	Accuracy (%)	COVID-19 accuracy (%)	Training (s)	Testing (s)
Luz et al. [17]	EfficientNet	93.9	–	–	–
Rahimzadeh and Attar [18]	Xception - ResNet50V2	91.4	99.6	–	–
Punn and Agarwal [20]	NASNetLarge	97.0	–	–	–
Khan et al. [21]	CoroNet (Xception)	89.5	96.6	–	–
Wang and Wong [22]	Tailored CNN	92.3	80.0	–	–
Narin et al. [23]	ResNet50	98.6	–	–	–
Hemdan et al. [24]	VGG19	90.0	–	2641.0	4.0
Apostolopoulos and Mpesiana [25]	VGG19	93.5	–	–	–
Loey et al. [27]	GoogleNet	80.6	100.0	–	–
Kumar and Kumari [28]	ResNet50-SVM	95.4	–	–	–
Bukhari et al. [29]	ResNet50	98.2	–	–	–
Abbas et al. [30]	DeTrac	95.1	–	–	–
Islam et al. [31]	CNN-LSTM	97.0	–	–	–
Faisal et al. [32]	VGG-19	99.5	100.0	–	–
Dey et al. [33]	Kapur's Entropy	90.0	–	–	–
Singh et al. [35]	SVM	99.6	–	–	–
Ucar and Korkmaz [60]	COVIDiagnosis-Net	98.3	100.0	2277.6	–
Asnaoui et al. [61]	Inception-ResNetV2	92.2	–	79184.3	262.0
Li et al. [62]	DenseNet	88.9	79.2	–	–
Chowdhury et al. [63]	Sgdm-SqueezeNet	98.3	96.7	–	–
Yang et al. [64]	VGG16	99.0	–	–	–
Suppakitjanusant et al. [65]	VGG19	85.0	–	–	–
Zhao et al. [66]	Bit-M	99.2	–	–	–
Sharmila et al. [67]	DCGANs	98.6	–	–	–
Reis et al. [68]	COVID-DSNet	97.6	–	–	–
Das et al. [69]	Ensemble method	91.62	–	–	–
Proposed system	**VGG19-RNN**	**99.9**	**99.9**	**16722.4**	**129.7**

References

[1] About Worldometer COVID-19 data - Worldometer. https://www.worldometers.info/coronavirus/ (Accessed 01 2023).

[2] Advice for the public. https://www.who.int/emergencies/diseases/novel-coronavirus-2019/advice-for-public (Accessed 01 2023).

[3] Everything about the Corona virus - Medicine and Health. (Accessed 01 2023).

[4] T. Ai, Z. Yang, L. Xia, Correlation of chest CT and RT-PCR testing in Coronavirus disease, Radiology 2019 (2020) 1–8, http://dx.doi.org/10.14358/PERS.80.2.000.

[5] C. Long, H. Xu, Q. Shen, X. Zhang, B. Fan, C. Wang, B. Zeng, Z. Li, X. Li, H. Li, Diagnosis of the Coronavirus disease (COVID-19): rRT-PCR or CT? Eur. J. Radiol. 126 (2020) 108961, http://dx.doi.org/10.1016/j.ejrad.2020.108961.

[6] H. Shi, X. Han, N. Jiang, Y. Cao, O. Alwalid, J. Gu, Y. Fan, C. Zheng, Articles Radiological findings from 81 patients with COVID-19 pneumonia in Wuhan, China: a descriptive study, Lancet Infect. Dis. 20 (2020) 425–434, http://dx.doi.org/10.1016/S1473-3099(20)30086-4.

[7] Z.Y. Zu, M. Di Jiang, P.P. Xu, W. Chen, Q.Q. Ni, G.M. Lu, L.J. Zhang, Coronavirus disease 2019 (COVID-19): A perspective from China, Radiology (2020) 200490, http://dx.doi.org/10.1148/radiol.2020200490.

[8] G.D. Rubin, C.J. Ryerson, L.B. Haramati, N. Sverzellati, J.P. Kanne, S. Raoof, N.W. Schluger, A. Volpi, J.-J. Yim, I.B.K. Martin, D.J. Anderson, C. Kong, T. Altes, A. Bush, S.R. Desai, J. Goldin, J.M. Goo, M. Humbert, Y. Inoue, H.-U. Kauczor, F. Luo, P.J. Mazzone, M. Prokop, M. Remy-Jardin, L. Richeldi, C.M. Schaefer-Prokop, N. Tomiyama, A.U. Wells, A.N. Leung, The role of chest imaging in patient management during the COVID-19 pandemic, Chest (2020) 1–11, http://dx.doi.org/10.1016/j.chest.2020.04.003.

[9] F. Shaheen, B. Verma, M. Asafuddoula, Impact of Automatic Feature Extraction in Deep Learning Architecture, 2016 Int. Conf. Digit. Image Comput. Tech. Appl.

DICTA 2016, 2016.
[10] A. Asraf, M.Z. Islam, M.R. Haque, M.M. Islam, Deep learning applications to combat novel Coronavirus (COVID-19) pandemic, SN Comput. Sci. 1 (2020) 1–7, http://dx.doi.org/10.1007/s42979-020-00383-w.
[11] P. Saha, M.S. Sadi, M.M. Islam, EMCNet: Automated COVID-19 diagnosis from X-ray images using convolutional neural network and ensemble of machine learning classifiers, Informatics Med. Unlocked. 22 (2021) 100505, http://dx.doi.org/10.1016/j.imu.2020.100505.
[12] Y. Guo, Y. Liu, E.M. Bakker, Y. Guo, M.S. Lew, CNN-RNN: a large-scale hierarchical image classification framework, Multimedia Tools Appl. 77 (2018) 10251–10271, http://dx.doi.org/10.1007/s11042-017-5443-x.
[13] Q. Yin, R. Zhang, X. Shao, CNN and RNN mixed model for image classification, MATEC Web Conf. 277 (2019) 02001, http://dx.doi.org/10.1051/matecconf/201927702001.
[14] T. Nakamura, T. Higuchi, H. Sawada, Phosphorylation of 6 mercaptopurine in leukemic cells, J. Japan Soc. Cancer Ther. 13 Th Cong (1976) 245–246.
[15] L.J. Muhammad, M.M. Islam, S.S. Usman, S.I. Ayon, Predictive data mining models for novel Coronavirus (COVID-19) infected patients' recovery, SN Comput. Sci. 1 (2020) 206, http://dx.doi.org/10.1007/s42979-020-00216-w.
[16] T. Mahmud, M.A. Rahman, S.A. Fattah, CovXNet: A multi-dilation convolutional neural network for automatic COVID-19 and other pneumonia detection from chest X-ray images with transferable multi-receptive feature optimization, Comput. Biol. Med. 122 (2020) 103869, http://dx.doi.org/10.1016/j.compbiomed.2020.103869.
[17] E. Luz, P.L. Silva, R. Silva, L. Silva, G. Moreira, D. Menotti, Towards an effective and efficient deep learning model for COVID-19 patterns detection in X-ray images, 2020, pp. 1–10.
[18] M. Rahimzadeh, A. Attar, A New Modified Deep Convolutional Neural Network for Detecting COVID-19 from X-Ray Images, 2020, http://arxiv.org/abs/2004.08052.
[19] S. Minaee, R. Kafieh, M. Sonka, S. Yazdani, G.J. Soufi, Deep-COVID: Predicting COVID-19 from Chest X-Ray Images using Deep Transfer Learning, 2020, http://arxiv.org/abs/2004.09363.
[20] N.S. Punn, S. Agarwal, Automated diagnosis of COVID-19 with limited posteroanterior chest X-ray images using fine-tuned deep neural networks, 2020, http://arxiv.org/abs/2004.11676.
[21] A.I. Khan, J.L. Shah, M. Bhat, CoroNet: A Deep Neural Network for Detection and Diagnosis of Covid-19 from Chest X-ray Images, 2020, http://arxiv.org/abs/2004.04931.
[22] L. Wang, A. Wong, COVID-Net: A Tailored Deep Convolutional Neural Network Design for Detection of COVID-19 Cases from Chest X-Ray Images, 2020, http://arxiv.org/abs/2003.09871.
[23] A. Narin, C. Kaya, Z. Pamuk, Zonguldak Bulent Ecevit University, 67100, Zonguldak, Turkey Department of Biomedical Engineering, 2020, ArXiv Prepr., arXiv:2003.10849 https://arxiv.org/abs/2003.10849.
[24] E.E.-D. Hemdan, M.A. Shouman, M.E. Karar, COVIDX-Net: A Framework of Deep Learning Classifiers to Diagnose COVID-19 in X-ray Images, 2020, http://arxiv.org/abs/2003.11055.
[25] I.D. Apostolopoulos, T.A. Mpesiana, Covid-19: automatic detection from X-ray images utilizing transfer learning with convolutional neural networks, Phys. Eng. Sci. Med. (2020) 1–8, http://dx.doi.org/10.1007/s13246-020-00865-4.
[26] M.J. Horry, S. Chakraborty, M. Paul, A. Ulhaq, B. Pradhan, X-ray image based COVID-19 detection using pre-trained deep learning models, 2007.
[27] M. Loey, F. Smarandache, N.E.M. Khalifa, Within the lack of chest COVID-19 X-ray dataset: A novel detection model based on GAN and deep transfer learning, Symmetry (Basel) 12 (2020) http://dx.doi.org/10.3390/SYM12040651.
[28] P. Kumar, S. Kumari, Detection of coronavirus disease (COVID-19) based on deep features, 9, 2020, http://dx.doi.org/10.20944/preprints202003.0300.v1, https://www.preprints.org/manuscript/202003.0300/V1.
[29] S.U.K. Bukhari, S.S.K. Bukhari, A. Syed, S.S.H. Shah, The diagnostic evaluation of Convolutional Neural Network (CNN) for the assessment of chest X-ray of patients infected with COVID-19, 2020, http://dx.doi.org/10.1101/2020.03.26.20044610, MedRxiv. 2020.03.26.20044610.
[30] A. Abbas, M.M. Abdelsamea, M.M. Gaber, Classification of COVID-19 in chest X-ray images using DeTraC deep convolutional neural network, 2020, http://arxiv.org/abs/2003.13815.
[31] M.Z. Islam, M.M. Islam, A. Asraf, A Combined Deep CNN-LSTM Network for the Detection of Novel Coronavirus (COVID-19) Us- ing X-ray Images, 2020, pp. 1–20, http://dx.doi.org/10.1101/2020.06.18.20134718.
[32] N.B. Prakash, M. Murugappan, G.R. Hemalakshmi, M. Jayalakshmi, M. Mahmud, Deep transfer learning for COVID-19 detection and infection localization with superpixel based segmentation, Sustain. Cities Soc. 75 (2021) 103252, http://dx.doi.org/10.1016/j.scs.2021.103252.
[33] N. Dey, V. Rajinikanth, S.J. Fong, M.S. Kaiser, M. Mahmud, Social group optimization–Assisted Kapur's entropy and morphological segmentation for automated detection of COVID-19 infection from computed tomography images, Cognit. Comput. 12 (2020) 1011–1023, http://dx.doi.org/10.1007/s12559-020-09751-3.
[34] V.N.M. Aradhya, M. Mahmud, D.S. Guru, B. Agarwal, M.S. Kaiser, One-shot cluster-based approach for the detection of COVID–19 from chest X–ray images, Cognit. Comput. 13 (2021) 873–881, http://dx.doi.org/10.1007/s12559-020-09774-w.
[35] A.K. Singh, A. Kumar, M. Mahmud, M.S. Kaiser, A. Kishore, COVID-19 infection detection from chest X-ray images using hybrid social group optimization and support vector classifier, Cognit. Comput. (2021) http://dx.doi.org/10.1007/s12559-021-09848-3.
[36] M. Shamim Kaiser, M. Mahmud, M.B.T. Noor, N.Z. Zenia, S. Al Mamun, K.M. Abir Mahmud, S. Azad, V.N. Manjunath Aradhya, P. Stephan, T. Stephan, R. Kannan, M. Hanif, T. Sharmeen, T. Chen, A. Hussain, IWorksafe: Towards healthy workplaces during COVID-19 with an intelligent phealth app for industrial settings, IEEE Access 9 (2021) 13814–13828, http://dx.doi.org/10.1109/ACCESS.2021.3050193.
[37] J.P. Cohen, P. Morrison, L. Dao, Covid-19 image data collection, 2023, (Accessed 01 2023).
[38] COVID-19 chest X-ray. https://github.com/agchung (Accessed 01 2023).
[39] Radiopaedia. COVID-19 X-ray Cases. 2023 (Accessed 01 2023).
[40] COVID-19 DATABASE / SIRM. https://www.sirm.org/en/category/articles/covid-19-database/ (Accessed 01 2023).
[41] COVID-19 Chest X-ray Image Repository. https://figshare.com/articles/COVID-19_Chest_X-ray_Image_Repository/12580328/2 (Accessed 01 2023).
[42] COVID-19 Image Repository. https://figshare.com/articles/COVID-19_Image_Repository/12275009/1 (Accessed 01 2023).
[43] Mendeley Data - Augmented COVID-19 X-ray Images Dataset. https://data.mendeley.com/datasets/2fxz4px6d8/4 (Accessed 01 2023).
[44] Chest X-ray Images (Pneumonia)/Kaggle, Kaggle. https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia (Accessed 01 2023).
[45] NIH Chest X-rays/Kaggle. https://www.kaggle.com/nih-chest-xrays/data? (Accessed 01 2023).
[46] M. Huh, P. Agrawal, A.A. Efros, What makes ImageNet good for transfer learning?, 2016.
[47] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: 3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc., 2015, pp. 1–14.
[48] A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks, Commun. ACM. 60 (2017) 84–90, http://dx.doi.org/10.1145/3065386.
[49] G. Huang, Z. Liu, L. Van Der Maaten, K.Q. Weinberger, Densely connected convolutional networks, in: Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017. 2017-(2017), 2017, pp. 2261–2269, http://dx.doi.org/10.1109/CVPR.2017.243.
[50] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, Rethinking the inception architecture for computer vision, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. 2016-, 2016, pp. 2818–2826, http://dx.doi.org/10.1109/CVPR.2016.308.
[51] U. Nazir, N. Khurshid, M.A. Bhimra, M. Taj, Tiny-Inception-ResNet-v2: Using Deep Learning for Eliminating Bonded Labors of Brick Kilns in South Asia, 2019, http://arxiv.org/abs/1907.05552.
[52] P.J. Werbos, Backpropagation through time: What it does and how to do it, Proc. IEEE. 78 (1990) 1550–1560, http://dx.doi.org/10.1109/5.58337.
[53] Yoshua. Bengio, Patrice. Simard, Paolo. Frasconi, Learning long-term dependencies with gradient descent is difficult, IEEE Trans. Neural Netw. 5 (2014) 157.
[54] S. Hochreiter, The vanishing gradient problem during learning recurrent neural nets and problem solutions, Int. J. Uncertainty, Fuzziness Knowlege-Based Syst. 6 (1998) 107–116, http://dx.doi.org/10.1142/S0218488598000094.
[55] P. Liu, X. Qiu, X. Chen, S. Wu, X. Huang, Multi-timescale long short-term memory neural network for modelling sentences and documents, in: Conf. Proc. - EMNLP 2015 Conf. Empir. Methods Nat. Lang. Process, 2015, pp. 2326–2335, http://dx.doi.org/10.18653/v1/D15-1280.
[56] M.J. Brown, L.A. Hutchinson, M.J. Rainbow, K.J. Deluzio, A.R. De Asha, A comparison of self-selected walking speeds and walking speed variability when data are collected during repeated discrete trials and during continuous walking, J. Appl. Biomech. 33 (2017) 384–387, http://dx.doi.org/10.1123/jab.2016-0355.
[57] G.E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, R.R. Salakhutdinov, Improving neural networks by preventing co-adaptation of feature detectors, 2012, pp. 1–18.
[58] D.M. Hawkins, The problem of overfitting, J. Chem. Inf. Comput. Sci. 44 (2004) 1–12, http://dx.doi.org/10.1021/ci0342472.
[59] D.P. Kingma, J.L. Ba, Adam: A method for stochastic optimization, in: 3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc., 2015, pp. 1–15.
[60] F. Ucar, D. Korkmaz, COVIDiagnosis-Net: Deep Bayes-SqueezeNet based diagnosis of the coronavirus disease 2019 (COVID-19) from X-ray images, Med. Hypotheses. 140 (2020) http://dx.doi.org/10.1016/j.mehy.2020.109761.

[61] K. El Asnaoui, Y. Chawki, Using X-ray images and deep learning for automated detection of coronavirus disease, J. Biomol. Struct. Dyn. (2020) 1–12, http://dx.doi.org/10.1080/07391102.2020.1767212.

[62] X. Li, C. Li, D. Zhu, COVID-MobileXpert: On-Device COVID-19 Screening using Snapshots of Chest X-ray, 2020, http://arxiv.org/abs/2004.03042.

[63] M.E.H. Chowdhury, T. Rahman, A. Khandakar, R. Mazhar, M.A. Kadir, Z. Bin Mahbub, K.R. Islam, M.S. Khan, A. Iqbal, N. Al-Emadi, M.B.I. Reaz, Can AI help in screening viral and COVID-19 pneumonia?, 2020, http://arxiv.org/abs/2003.13145.

[64] D. Yang, C. Martinez, L. Visuña, H. Khandhar, C. Bhatt, J. Carretero, Detection and analysis of COVID-19 in medical images using deep learning techniques, Sci. Rep. 11 (2021) 1–13, http://dx.doi.org/10.1038/s41598-021-99015-3.

[65] P. Suppakitjanusant, S. Sungkanuparph, T. Wongsinin, S. Virapongsiri, N. Kasemkosin, L. Chailurkit, B. Ongphiphadhanakul, Identifying individuals with recent COVID-19 through voice classification using deep learning, Sci. Rep. 11 (2021) 1–7, http://dx.doi.org/10.1038/s41598-021-98742-x.

[66] W. Zhao, W. Jiang, X. Qiu, Deep learning for COVID-19 detection based on CT images, Sci. Rep. 11 (2021) 1–12, http://dx.doi.org/10.1038/s41598-021-93832-2.

[67] S.V.J., J.F.D., Deep learning algorithm for COVID-19 classification using Chest X-ray images, Comput. Math. Methods Med. 2021 (2021) 9269173, http://dx.doi.org/10.1155/2021/9269173.

[68] H.C. Reis, V. Turk, COVID-DSNet: A novel deep convolutional neural network for detection of coronavirus (SARS-CoV-2) cases from CT and Chest X-ray images, Artif. Intell. Med. 134 (2022) 102427, http://dx.doi.org/10.1016/j.artmed.2022.102427.

[69] A.K. Das, S. Ghosh, S. Thunder, R. Dutta, S. Agarwal, A. Chakrabarti, Automatic COVID-19 detection from X-ray images using ensemble learning with convolutional neural network, Pattern Anal. Appl. 24 (2021) 1111–1124, http://dx.doi.org/10.1007/s10044-021-00970-4.

BenchCouncil Transactions on Benchmarks, Standards and Evaluations 2 (2022) 100085

Contents lists available at ScienceDirect

BenchCouncil Transactions on Benchmarks, Standards and Evaluations

journal homepage: www.keaipublishing.com/en/journals/benchcouncil-transactions-on-benchmarks-standards-and-evaluations/

Review Article

An extensive study on Internet of Behavior (IoB) enabled Healthcare-Systems: Features, facilitators, and challenges

Mohd Javaid [a,*], Abid Haleem [a], Ravi Pratap Singh [b], Shahbaz Khan [c], Rajiv Suman [d]

[a] Department of Mechanical Engineering, Jamia Millia Islamia, New Delhi, India
[b] Department of Mechanical Engineering, National Institute of Technology, Kurukshetra, Haryana, India
[c] Institute of Business Management, GLA University, Mathura, UP, India
[d] Department of Industrial & Production Engineering, G.B. Pant University of Agriculture & Technology, Pantnagar, Uttarakhand, India

ARTICLE INFO

Keywords:
Internet of Behavior (IoB)
Healthcare
Patient
Behaviour

ABSTRACT

The Internet of Behaviour (IoB) is an effort to dissect behavioural patterns as explained by data collection. IoB is an extension of the Internet of Things (IoT). Therefore, both are anticipated to experience exponential growth in the upcoming years. Healthcare firms have many opportunities to employ IoB to provide individualised services and anticipate patients' behaviour. As behaviour and analysis are closely related to psychology, many techniques exist to collect relevant data. The IoB improves the doctor's and patient's experience. As IoT and IoB are interconnected, IoB technology collects and analyses data depending on user activity. These offer a practical technique for developing real-time remote health monitoring systems. This technology aids in the optimisation of auto insurance premiums in the healthcare sector. It tries to alter patient behaviour in order to improve the treatment process. IoB has applications in various areas, including retail and entertainment, and has the potential to change the marketing sector significantly. This technology is helpful for the appropriate analysis and comprehension of behavioural data used for creating valuable services for treatment. The primary purpose of this paper is to study IoB and its need for healthcare. The working process structure and features of IoB for the healthcare domain are studied. This paper further identifies and analyses the significant applications of IoB for healthcare. In the future, IoB technologies will give us a higher quality of life and well-being. IoB is the ideal fusion of technology, data analytics, and behavioural science. This will help healthcare professionals collect data and analyse the patient's behaviours for an efficient treatment process. The IoB will be the digital ecosystem's intelligence in a few years.

1. Introduction

The Internet of Things (IoT) leads towards the Internet of Behavior (IoB). IoT's typically referred to as a network of physical items implanted with sensors, software, and other technologies to connect and exchange data with other systems and devices over the Internet. IoB is considered one of the cutting-edge technologies that the world is currently experiencing. The IoB gathers information on how devices are used to learn more about user behaviour, interests, and preferences. IoB attempts to comprehend user online activity data from a human psychology perspective. This technology assists in identifying patterns and recommendations for customer behaviour [1–3]. Companies are now capable of knowing customer needs. IoB data collection will aid in understanding consumer behaviour and patterns. IoB firms aid in campaign optimisation and enhance client satisfaction. This technology collects data from sensors, devices, geo-tagging activities, cookies, browser histories, social media activity, and other sources [4,5]. These data are used for several analyses to predict consumer behaviour and needs.

Data security issues are common and may expose personal data such as health status or medical history. Identity theft, online fraud, and technology theft are increasingly common in the IoT-enabled ecosystem. It is also possible to get sensitive information like delivery routes and banking codes using IoB. Businesses may use deception to get customers to spend more money on certain products. Using consumer data or private information may give rise to privacy problems due to a need for more data regulation in the online sphere. Customers will benefit from a personalised experience using IoB. As a result, relevant information, offers, prices, discounts, and more will be displayed along with suggested adverts. In order to interact with clients in real time, the IoB is considered an essential tool. It accomplishes this by offering

* Corresponding author.
E-mail addresses: mjavaid@jmi.ac.in (M. Javaid), ahaleem@jmi.ac.in (A. Haleem), singhrp@nitkkr.ac.in (R.P. Singh), shahbaz.me12@gmail.com (S. Khan), dr.r.suman@gbpuat-tech.ac.in (R. Suman).
URL: https://scholar.google.co.in/citations?user=rfyiwvsAAAAJ&hl=en (M. Javaid).

https://doi.org/10.1016/j.tbench.2023.100085
Received 15 December 2022; Received in revised form 28 January 2023; Accepted 28 January 2023
Available online 2 February 2023

pertinent information about the customer while they are preoccupied with their product or service search [6–8].

Companies and organisations that successfully navigate the difficulties of establishing and maintaining an IoB system might benefit. The major challenge in IoB adoption is gathering private information from customers and employees. People are reluctant to give some personal information for ease and other advantages. Businesses can learn more about their customer's preferences and buying behaviours based on how they use various platforms and devices [9,10]. Therefore, customers will be able to solve their problems quickly, resulting in a pleasant customer experience. We can predict and enhance individual behaviour using IoB. Leading companies have admitted to exchanging customer information with other companies without the customer's consent. IoB establishes a digital link between people's behaviour and activities, allowing precision targeting and delivering information and services to affect behaviour. When it comes to addressing individual and situational needs, it is intended to secure fast, relevant, and accurate communication, offerings, and services better than any Artificial Intelligence (AI)/Machine Learning (ML)-based system can [11,12].

Digital transformation technologies significantly impact several industries, and healthcare is one of them. The IoB is gradually displacing the discussions around IoT development, AI, and robotics technologies utilised in the healthcare sector. The IoB will be utilised in the healthcare sector to track patient behaviour and related activities regarding disease and its treatment. They will be monitored to ensure the task is completed on the prescribed schedule [13,14]. IoB is more than simply data analysis; when combined with effective digital marketing methods, it may significantly increase medical product sales with digital marketing. Users of digital tracking do not object when the information supplied gives value to their daily lives, making it the best choice to research market trends before introducing a product. This technology can also persuade a consumer to purchase a product. For instance, smartwatches that monitor a user's heart rate, blood pressure, sugar levels, water intake, physical activity, and other data analyse lifestyle errors and recommend a better way of living. The IoB requires an internet connection to function; therefore, digital services will be among the healthcare that benefits most from this technology [15–17]. Therefore, this study discusses the significant capabilities of IoB in healthcare.

2. Internet of behavior

The term "Internet of Behavior" refers to the gathering and application of data to influence behaviour. These data are gathered by electrical appliances, personal internet activities, and wearable technology, and they can provide essential details about user behaviour and preferences. IoB is an extension of IoT that entails using data gathered from IoT devices to use feedback loops to affect customer actions and behaviours. It is based on an understanding of human psychology, such as making purchases, adhering to a particular online brand, or tracking and analysing those behaviours utilising smart technology and machine learning algorithms. The IoB is a body of data that contains vital information on consumer behaviours, interests, and preferences. The IoB attempts to comprehend data obtained from users' online behaviours from a behavioural psychology standpoint. It aims to solve the issue of how to comprehend data and utilise that knowledge to develop and advertise new goods from the perspective of human psychology [18–20].

The "IoB" refers to a method for user-controlled data analysis based on behavioural psychology. The research has an influence on user experience, search experience optimisation, and how final products and services are marketed and promoted by a corporation. In addition to technical complexities, IoB is also challenging psychologically. For ethical and regulatory reasons, it is essential to undertake statistical studies that record usual routines and behaviours without entirely compromising client privacy. Data analytics, behavioural science, and artificial intelligence are all combined to study human behaviour through data mining. IoB seeks to find methods to turn knowledge gleaned from internet user activities into something useful. Because it may provide a thorough and customised knowledge of the clients, IoB has the potential to become an efficient marketing tool for businesses all over the globe. This technology is helpful for the healthcare industry in delivering individualised care [21–23].

3. Need of IoB in healthcare

There is a need for IoB in healthcare to perform daily operations, including treatment planning, operations scheduling etc. IoB assists in determining the primary influencing aspects of a patient's behaviour. IoB also assists buyers in obtaining their desired services without wasting time navigating various purchasing methods for healthcare. Additionally, this technology can assist firms in developing goal-driven plans to delight clients and increase sales rates by analysing data [24,25]. This innovation will fundamentally alter consumer purchasing behaviour and could revolutionise how goods are purchased. Many users are happy to share their personal information, even though some are hesitant to do so unless it adds value to their treatment services. Data from every aspect of a user's life may be gathered to improve performance and quality. This enables numerous touchpoints for the customer to interact with [26,27].

A large amount of data that can influence or drive patient behaviour are gathered through IoB. By examining online user behaviour, it seeks to comprehend user psychology. This framework can gather, examine, comprehend, and react to various human activities through machine learning algorithms. Many firms have been able to use online advertising to reach more clients by implementing IoB technologies. Businesses may quickly identify and target particular people or groups to offer their services and products using IoB. For instance, Google and Facebook use behavioural data to show their consumers relevant advertisements. With IoB, businesses may track customer behaviour to provide better services while connecting with potential customers [28–30].

4. Research objectives

The goal of IoB is to predict and alter behaviour using data. IoB implementation varies depending on the industry. Real-time workload and delivery schedule management are significant benefits of IoB [31,32]. IoB prioritises gathering, analysing, and comprehending user behaviour to enhance the value chain and service quality. With the help of this technology, behavioural science can provide more significant insights from the data. Additionally, as IoB ensures two-way communication with clients, it aids in improving customer relationships. Instead of doing surveys to gather feedback from clients, businesses can much more effectively identify their needs and offer a beneficial upgrade [33–35]. The primary research objectives of this article are as under:

RO1: - to study IoB and identify its need for healthcare;

RO2: - to study the working process structure of IoB for the healthcare domain;

RO3: - to discuss the various considered features of IoB for the healthcare sector;

RO4: - to study and identify major applications of IoB for healthcare.

5. Working process structure of IoB for the healthcare domain

The process of IoB started with the fundamentals of IoT in terms of data flow and information sharing. This concept has gained attention while targeting the improvement in serving the customers of the numerous services. Fig. 1 depicts the process and working structure of IoB philosophy towards updating and supporting the healthcare sectors. Knowledge with improved and enhanced wisdom is an integral part of this process flow of the IoB theme [36–38].

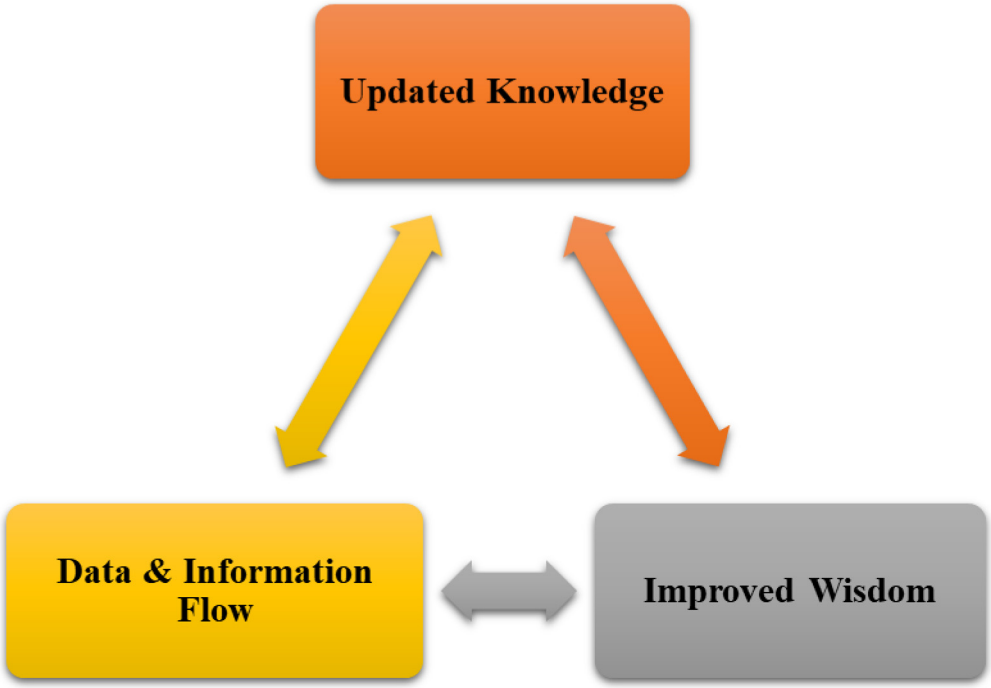


Fig. 1. Smart process structure of IoB for healthcare.

The IoB helps users, particularly those working in retail, healthcare and consumer industries, to understand the demands and preferences of their customers. IoB, applications help observe consumer needs and preferences, especially organisations, to navigate the crisis with minimal collateral damage if it understands its target audience. As a result, the introduction of the internet and social media marketing has substantially changed how client behaviour is approached and studied. This brings us to the IoT, a network of connected devices used for data management and sharing [39–41]. The IoT helps turn data into information, whereas IoB translates this information into valuable knowledge. Many companies have shifted to using social and digital media to sell their products during the pandemic. The IoB is considered one of the crucial tools for organisations looking to improve their online visibility [42,43].

Although the IoB is still developing, it has numerous advantages for healthcare. It aids marketers and business owners in thoroughly understanding their target market. Understanding online behaviour will improve the client experience. The IoB is concerned with gathering, analysing, and utilising information on human behaviour to modify it positively. With the emergence of social media, consumer analytics and targeted advertising have increased dramatically [44,45]. Companies may now easily reach their target audience where they are already spending time through social media. The popularity of IoT devices will undoubtedly encourage the use of IoB, much as the smartphone boom contributed to the growth of social media. Cybercriminals may be able to acquire private information about consumer preferences and gather market access codes, delivery routes, and bank codes using behavioural data. The IoB has the potential to be an effective new sales and marketing tool for companies and organisations. Digital marketing is among the many fields and ways of doing business that is being radically altered [46–48].

Based on information gathered from various social media and other platforms, the IoB investigates consumer behaviour. The information gathered will be used to make assumptions about the lifestyle of the consumers. These gadgets provide online suggestions to users about services and products. It is also helpful to examine the car's speed, braking, acceleration, and other factors to determine how cautious the insured driver is using the data from IoT tracking devices [49–51]. The corporation lowers the premium the customer must pay after collecting data for a predetermined time if the user's behavioural data demonstrate fewer risk characteristics. The same information can be used to enforce safe driving habits and to evaluate a claim in the event of an accident. With the aid of IoB, healthcare may revolutionise its operational efficiency. In actuality, it was the industrial firms that popularised IoB during the Covid-19 pandemic [52,53].

Several protocols are developed and applied in the current healthcare industry, and these protocols are well adopted with the help of IoB. For instance, employees and visitors are continuously observed by computer vision to ensure that they are wearing masks properly. The other protocols' compliance was also checked using similar sensors. Therefore, automated alerts are sent anytime there seems to be a safety breach to guide them to the proper behaviour. IoB applications can guarantee better working conditions, increased productivity, and increased employee satisfaction in different sectors, including healthcare. The IoB offers valuable information on client behaviours, interests, and preferences [54,55]. The IoB aims to get user online activity data from a behavioural psychology standpoint. It addresses the issue of how to comprehend the data and use that information to develop and advertise new products from the human psychology perspective. IoB is a valuable method for analysing user-controlled data from behavioural psychology. Although IoB technology combined with IoT-harvested data can be used to market, not all of it is focused on advertising. Organisations will be able to evaluate, for instance, the effectiveness of both their for-profit and nonprofit efforts. Healthcare professionals can also track their efforts to engage and activate patients to improve their health [56,57].

6. Various considered features of IoB for the healthcare sector

Fig. 2 explores the several considered smart features and traits of IoB practice towards strengthening the healthcare systems. As reflected in Fig. 2, this concept involves the precise flow of information, which is further evolved with smart connectivity for processing patients' data effectively. This process is digital in its procedure and becomes quicker and faster, which ultimately results in empowered patients [58,59].

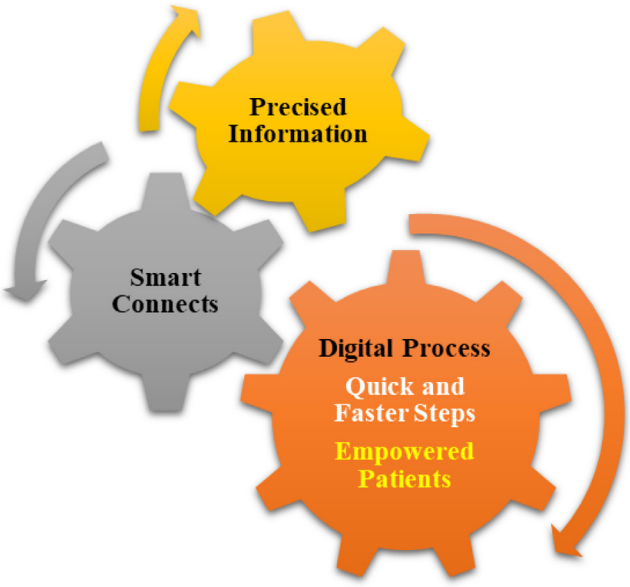


Fig. 2. Different considerations of IoB for healthcare systems.

Technology has advanced to the point that the term "IoB" uses data gathered by the IoT. First, IoB is applied to assessing adherence to health practices. IoB delivers comprehensible and practical advantages. Cybercriminals are particularly capable of utilising behaviour data. As a result, businesses need to be more alert and proactive in their data protection and maintaining privacy. Businesses gather and examine data for several purposes. This comprises, among others, guiding user experience design, generating products and services, aiding businesses in making informed business decisions, and tailoring marketing strategies [60,61]. Another element of the IoB is combining and evaluating data from various sources to reach a better and more efficient decision. Integrating IoT and IoB is seen as promising digital technologies that will soon become more practical in several domains. Even at an early stage, the benefits and potential of IoT solutions when integrated with IoB technology are clear. Even though IoB is still in its infancy, the IoT and its extensions will become essential in people's lives in the upcoming years, making life much simpler and more effective. It may develop into an ecosystem that defines the attitudes and behaviours that govern the digital world [62,63].

The world is now digitally capable of performing businesses and daily activities. The daily upgrades brought on by digital changes completely alter how businesses operate, and people live. It is also among the most popular technologies because it has fundamentally altered how devices are connected. IoB interprets information in light of particular human behaviours, such as purchasing habits and demographic interests. The IoB interprets the data gathered by IoT in conjunction with certain human behaviours, such as purchasing trends and community interests. Customers' behaviour when using maps is mainly influenced by devices connected to geolocation, big data, and facial recognition. IoB relies heavily on these data, yet it hides how consumers' data is gathered. Tracking their geographic location may determine whether someone has visited a store and how long they stayed there. Accordingly, companies can send out marketing messages, offers for promotions, and discounts to boost sales and give customers an outstanding shopping experience [64–66].

The IoB is used to modify how product marketing is done and obtain a fresh perspective on search experience optimisation or building design using the results of data analysis. The IoT is used with IoB since all data gathered from IoT and other sources are utilised to influence consumer behaviour. IoT technology gathers much data about interests and how to use items by connecting a phone with a laptop, voice assistant, or smart home. The tourism sector was another area where IoT and IoB significantly impacted [67,68]. Applications for making reservations can learn from past searches and other indexes, including demographic or social status data. It provides the most appropriate travel advice for customers. Data from numerous sources is analysed using e-commerce to learn more about customer behaviour. IoB enables researchers to understand how consumers first gained interest in a product and what factors influenced their purchasing decisions. This technology offers industry insights that enhance the proposals for the customers' demands [69,70].

Logistic IoB can be used for delivery planning, route, and correct route recommendations based on real-time data from various sensors. Telematic solutions are examples of the IoB. For instance, managers might use sophisticated car data to plan strategic routes. Additionally, this information may include details about a driver's behaviours, current information about accidents along a route, the sort of delivery to determine the most specific course, and logistics. The world is currently experiencing the emergence of the IoB and the IoT. The IoT and other sources are utilised to gather this data, which is then put to good use. It could provide insightful data on user preferences and behaviour. Gathering, analysing, understanding, and responding to various behaviours is the main objective of the IoB, which aims to improve the customer experience. Additionally, behavioural data enables firms to make better decisions regarding customer preferences and their experience. It also improves the value chain and the quality of the services. There are several locations where consumers may get information [71–73].

The IoT has now been expanded to include the IoB, where information is gathered from several connected devices to gain insightful knowledge about client interests, behaviours, and preferences. IoB aids in collecting, understanding, assessing, and assembling all forms of human behaviour. This aids in comprehending recent advances in technology and ML techniques. IoB is a potent instrument for boosting sales and developing exciting marketing campaigns. The IoB has attracted attention from all across the world. The IoB, like the IoT, may profoundly impact how people live. This technology can open up new technological frontiers [74,75].

7. IoB applications for the healthcare sector

IoB offers predictive data on any objectives and plans relevant to the current circumstance in healthcare. The IoB transforms into a tool for precise forecasting when it has significant users. This sets it apart from other apps that seek to track people's movements, find their locations, identify their faces, and determine their proximity to one another. However, combining these strategies can provide a very potent, situationally intelligent service. Modern tracking apps can incorporate IoB to capture the users' location [76–78]. The complexity of IoB is continually growing and changing, including how devices are connected, what calculations they can perform on their own, and how data is stored in the cloud. The transition to mobile devices has altered how individuals interact with one another and the outside world. The IoB devices' usage data provide valuable details on users' interests, actions, and preferences. Healthcare professionals can suggest a behaviour change programme for those conditions that can be prevented while IoB technology tracks progress along the way. Early detection via linked devices enables medical practitioners to start treatment earlier, even for non-preventable diseases; this relieves pressure on health systems and prolongs patient lives [79–81]. Table 1 discusses the significant applications of IoB for healthcare.

IoT devices collect usage and behavioural data, which offers insights into users' behaviours, interests, and preferences. Businesses are consequently putting more emphasis on IoB to collect such data for marketing and advertising purposes. The only real differentiator in today's commoditised world is in services, and IoB empowers businesses with superior servicing capabilities [82–84]. Both online and

Table 1
IoB application areas for the healthcare sector.

S No	Applications	Description
1.	Health-tracking	IoB is used to create health-tracking smartphone apps that measure a user's food intake, blood sugar levels, heart rates, and sleep patterns. However, they do have big plans for IoB technology. They want to make it simple to keep track of how they behave throughout the treatment. Companies that have access to the data IoT provides about us can now use IoB data to affect our behaviour. The app can alert to potentially hazardous situations and suggest behavioural modifications that result in a more advantageous or desirable outcome. A company's website, social media profiles, sensors, telematics, beacons, health monitors, and several other devices are a few of the places where consumer data may be obtained. IoB provides businesses in a wide range of sectors with innovative ways to sell their goods and services, enhance the value of their offers, and affect consumer and employee behaviour. Based on the data gathered, the technology enables them to increase the value of their relationships with clients and suppliers and improve financial results. Understanding behaviours through data will become an exciting component of every business as new IoT devices proliferate.
2.	Healthcare insurance	The adoption of IoB can benefit healthcare insurance as well. In particular, analysing consumer behaviour using data from IoT devices can aid in a more accurate estimation of insurance costs. Additionally, health insurance providers can utilise IoB to monitor clients' physical activity levels and determine how much to charge for premiums. IoB has the potential to be very helpful in the health insurance sector. Insurance firms monitor and secure motorist behaviour using driver-tracking software. Using IoB, they may assess the behaviour and decide whether a specific event resulted from an accident or an insured's mistaken assumption. Insurance providers now have a new potential to offer customised rates based on user-driving behaviour using the IoB. IoT devices can track the speed and distance of a car, typically used for driving, and offer the appropriate premium insurance. IoB aids in locating the target auditory that is the most precise. This is the fundamental tenet of the algorithms used by those businesses to guarantee that customers receive pertinent information. Monitoring straightforward visible actions by current digital technology has clear benefits for entertainment, sports, health, and life-coaching apps. The behavioural loop is incorporated right into Spotify, which is interesting since it allows users to communicate their desires and be rewarded with appropriate music whenever they want. IoB might also be used there to achieve a lot more.
3.	Determine health procedures	The role of the IoB is introduced in healthcare. In order to determine if health procedures are being followed during the continuing COVID-19 pandemic. Furthermore, due to the COVID-19 pandemic, tiredness, a condition when individuals relax their adherence to public health precautions, the usage of IoB in medical devices, for this reason, will become more crucial than ever. Additionally, IoB uses thermal imaging to help detect people who have a fever. IoB can be a potent tool for combining sales and marketing to develop strategies that improve the products and services given to customers. IoB, for instance, is helpful in the medical industry since it enables medical professionals to evaluate patients' illnesses, responses to medicines, and other information about their way of life. Most location-based services track the user's location and send emails or notifications according to the GPS functionality of their mobile device or other methods like Bluetooth and near-field communication. Additionally, gathering information in real-time rather than after a delay aids businesses in making quick modifications/updations to their product offers.
4.	Assess patient activities	Healthcare practitioners may assess the extent of patient activity and participation. IoB may be used to assess how well healthcare activities are working. Organisations can use it to monitor staff compliance with pandemic health precautions such as mask wear, fever testing, and hand washing. Healthcare practitioners may also employ smart devices to monitor people's activities or whereabouts to ensure they are lowering their risk of contracting the virus by following social restrictions. IoB is, therefore, relevant to the well-being of the populace as a whole. IoB offers other advantages like a better comprehension of how consumers engage with items, improved insight into buying habits, real-time help, and customer communication in previously impractical ways. The IoB concept also centres on the appropriate analysis and comprehension of behavioural data and the aim to use knowledge to develop and market personalised goods and services that will be more valuable to customers and enterprises. Businesses and other organisations are heavily utilising this technology to increase their profits. By offering goods and services that are more suited to their needs and preferences, the IoB will also benefit customers.
5.	Wearing mask detection	Many computer vision businesses started employing IoB to detect whether citizens wore masks during the outbreak. Thermal fingers were employed to identify patients with elevated body temperatures in the same instance. IoB is helpful for face recognition to identify its customers' gender, age, and mood. In tailored advertising, the same system can perform admirably. Many digital marketing organisations currently employ analytics software to learn about consumer behaviour. The IoB allows marketers to reimagine the value chain, access previously inaccessible data, evaluate customer purchase patterns across platforms, and even deliver customised adverts and real-time point-of-sale notifications. Businesses can gain a deeper understanding of consumers' opinions towards particular goods or services, making it even simpler to address customer complaints. Sensors and RFID tags are already being used by businesses in the manufacturing sector to monitor how frequently on-site workers wash their hands. Additionally, computer vision can identify whether or not workers are adhering to social distancing instructions or mask procedures. Healthcare professionals can monitor patients' efforts to engage and activate.
6.	Disease surveillance	IoB is an effective tool for many different sectors. Disease surveillance, targeted shopping advice, car tracking for insurance, and fleet management are all made possible with this technology. People can use IoB applications to increase their effectiveness and satisfaction with goods and services. IoB focuses on using data analytics to change people's behaviour. Examples of the IoB applications that are ingrained in our daily lives include sensor-led driver assistance systems that advise safe driving techniques and health apps on our phones that track our diet, sleeping patterns, heart rate, blood glucose levels, etc. and suggest "habit alterations." Homes can become incredibly smart by using IoB. Earlier, we could use a smartphone or tablet and an Internet connection to remotely control appliances, thermostats, lighting, and other devices. Now, all the functionality of smart homes will automatically adapt to our preferences using information about our behaviour patterns previously collected by our devices. Though IoB has a good impact on our lives because it guides us in many areas, we should be aware that the system collects personal data and that the businesses that hold it bear much responsibility. The IoB concept entails changing our cultural norms and regulations established before the Digital Ages to transform our data into valuable knowledge about our decision-making patterns.
7.	Fitness tracking	Fitness tracker data is currently being used extensively to advance the healthcare sector. Big data is reviving the interaction between the sectors of health and fitness, two related fields. The industry benefits from all the information about people's lifestyles, health practices, fitness levels, and diets. However, users also benefit from the reminders and encouragement provided by the notifications that fitness trackers send out regarding things like calories, heart rate, blood pressure, and sleeping patterns. IoB devices are improved with embedded software and various sensors that collect data produced by people. Sensor data can be saved and analysed on a device or the cloud, depending on the device's computing power. Wireless body area networks can incorporate intrusive and wearable IoB device networks, which can be hybrid or wireless. IoB systems can safely exchange data in real time or at predetermined intervals with a central hub using connectivity technology. While the sensor data gathered by a smartwatch can offer insightful information about a patient's past, present, and future health concerns, the IoT also provides several non-invasive and highly effective diagnostic techniques. A stylish wearable device worn at the bottom of the ribcage can track lung function and detect early anomalies. Additionally, doctors can diagnose genetic abnormalities and treat diabetes more effectively by analysing data gathered by wearable sensors.

(*continued on next page*)

Table 1 (*continued*).

S No	Applications	Description
8.	Better healthcare solutions	The applications of IoB apply to every industry and sector. For instance, doctors might recommend better healthcare solutions to patients using data from wearables or healthcare applications. Patients with serious illnesses particularly benefit from it since IoB can monitor and spot irregularities, alerting concerned parties in real time. IoB hazards and advantages might be thoroughly researched and made publicly available as a solution to address various issues while removing marketing hype. In addition to addressing data privacy concerns, closer cooperation between regulators and device manufacturers may also lower the cost of IoB products. The IoB provides crucial information about consumer behaviour, interests, and passions. IoB uses behavioural psychology to try to make sense of the information gathered from user interactions online. IoB also uses available personal technologies like face recognition, location monitoring, and Big Data. As a result, it combines technology, data analytics, and behavioural psychology elements. The IoB's goal is to record, examine, comprehend, and react to all human activity in a way that enables tracking and interpreting of people who use cutting-edge technologies and advances in machine learning techniques.
9.	Analyse physiology	Most trackers or wearables make recommendations based on a comparison of baseline readings. The IoB can help when it comprehends the person's physiology thoroughly. IoB can assist with additional research in addition to providing particular recommendations. IoB allows for more excellent data collection and analysis of human behaviour in the actual world. IoB eliminates uncertainty about improving customer experience, increases the effectiveness and calibre of marketing campaigns, and alters how businesses interact with their customers by giving them priceless insights into user behaviour and the psychological factors involved in decision-making. Businesses can innovate to grow their companies by utilising the IoB. IoB will undoubtedly be at the forefront of creating new experiences for the global community. Currently, organisations use IoB to assist direct behaviour towards the desired results. It could be used to promote desired behaviour at work. For instance, firms may utilise computer vision to determine whether personnel are wearing masks or thermal imaging to monitor a rise in body temperature to ensure that the current health procedures are being followed. Similarly, sensors and RFID tags can track other hygiene practices like hand washing and space sanitisation.
10.	Analyse health conditions	Networked devices keep an eye on a person's health, gather physiological, biometric, or behavioural data, and communicate with one another across a wireless or hybrid network. The IoB cohort can also include independent mobile applications that examine physical activity and health-related information, including heartbeat, blood pressure, and sleep patterns. The IoB involves gathering, analysing, and interpreting data from IoT devices to spot trends in user behaviour and use this understanding to trigger specific behavioural events. These efforts improve business outcomes, such as increasing sales by effectively communicating with the relevant audience. Deeper behavioural insights are available as more devices are online. Additionally, businesses are likely to reward customers financially for opening up about their habits, way of life, preferences, and even dreams. IoB eliminates the requirement to create a perfect user persona. Big data enables the examination of prospective clients from multiple angles. One can create an extraordinarily detailed map of their customer's journey, use a highly tailored strategy, and add more points of contact. Users will use voice interaction with gadgets more frequently, moving in the direction of natural language and intent-based search.
11.	Customised treatment and medication	IoB devices could aid medical personnel in spotting repeating trends in patient data and developing customised treatment and medication regimens catered to the requirements of a particular patient or patient group. For this reason, electronic health records could be enhanced with sensor data and subjected to artificial intelligence algorithms analysis. Health insurance businesses can adopt a more detailed approach to risk profiling and optimise insurance policies based on a person's medical history, occupation, and lifestyle. IoB devices can measure several bodily data, such as cardiac rhythms, sleep patterns, menstrual cycles, and users' whereabouts. IoT creates a lot of information and data. It is simple to identify the platforms that consumers engage with and obtain comprehensive information about clients after IoB processes this previously unavailable information. These users'/customers' data can be used to develop efficient marketing strategies, after which real-time notifications and targeted advertisements can be sent to them. Businesses can utilise the comprehensive data gleaned from the IoB to enhance the broad product experience for clients. In order to gather information on individual behaviour and cognitive patterns, the IoB combines IoT, behavioural science, and data analytics. This data is analysed to learn more about behaviour used for various things, such as enhancing marketing campaigns or patient medical monitoring.
12.	Track daily activities	A health application that can keep track of a person's diet, exercise, weight, sleep patterns, heart rate, stress level, oxygen level, blood sugar levels, and similar factors can notify the user so they can seek help or advice from healthcare professionals and work towards a positive outcome using IoB. This also records information about a driver's driving habits. In order to achieve the desired goal of selling their goods and services, businesses are now leveraging the IoB to track changes in behaviour. This cutting-edge technology can help businesses in various ways, such as by improving customer interaction, detecting when customers are interested in particular products and collecting more significant insights into the user journey from discovery to purchase. Healthcare delivery will change as a result of IoB technology. More information about connected devices may be provided. Behavioural analytics give clinicians even more information to forecast chronic diseases and other health conditions and take preventative action. Predictive technologies can quickly spot trends of lifestyle behaviour or early signs of sickness by monitoring an individual's real-life behaviour through IoB.

physical movements can be monitored on a smartphone. These days many people connect our smartphones, computers, voice assistants, home and car cameras, and in the case of a smart home, practically all of the interior equipment. Our likes, dislikes, lifestyle, interests, favourite restaurants, favourite apparel stores, trip plans and locations, travel duration, purchasing habits, and much more can be revealed to a firm by this combined with the social data from our social media footprint. IoT-enabled vehicles are gaining popularity and transmitting information about drivers' driving habits [85–88]. IoT and IoB can undoubtedly deliver data-driven value that is utilised by industries. Banks can now persuade us to save more; auto insurance providers can drive safely and benefit from low premiums; health insurance providers can persuade us to lead healthy lifestyles and benefit from low premiums etc. [89,90].

In practically every business, the IoB redesigns the value chain in addition to having an impact on consumer choice. Because the IoT deals with personal data, it enters the grey legal region where it does not meet the current standard. The IoB links data to human behaviour, and decision-making companies will continuously track our activities using the massive amounts of data they previously collected from internet-connected smart products. Businesses have successfully connected all significant equipment to the internet, making it simple for them to keep on their watchlist [91–94]. IoB integrates already-existing technologies that target the individual, like facial recognition, location tracking, and big data. The goal of IoB is to record, examine, comprehend, and react to all sorts of human behaviour in a way that enables tracking and analysing those behaviours utilising developing machine learning algorithms and emerging technological advancements. The software company has created a health app for cell phones that monitors blood sugar levels, heart rate, sleep habits, and food. The software can notify the user when their health is in danger and offer behavioural suggestions for a better outcome [95–98].

8. Discussion

IoB interprets the information gathered by IoT and links it to unique human behaviours, like selecting a particular brand. IoB transforms the data and information gathered by IoT into knowledge and possibly wisdom for societal benefits. Technology, data and analytics, and behavioural science are all combined in IoB. Data is extracted using technology, and information is drawn out of the data using analytics.

Personalisation is one of the most crucial elements of any successful service. The right mix of pertinent consumers makes the business more successful. The idea behind the IoB is to transform data into insightful knowledge about various user preferences that can be used as a standard for forecasting consumer behaviour. The system determines which psychological factors affect to get a particular result. This opens up a wide range of fresh marketing strategies to attract more clients or advertise a particular product. Additionally, it makes marketing campaigns more focused, which results in more effective advertising and delighted customers.

IoB also aids in the elimination of numerous specialised studies and surveys that incur a tremendous amount of cost. Several apps can efficiently gather and analyse any information that is still available on the internet. Numerous applications already exist that evaluate user behaviour using information from gadgets and provide recommendations aimed at helping users adopt a healthier lifestyle. For a psychologist, it seems natural to classify behaviour as including things like planning, emotional experiences, interpretations, and goals. All these human occurrences are intended to be covered and "coded" by IoB. Most digital tools and apps that track our behaviour, such as while chatting, travelling, visiting locations, using services, and exercising, record the person's identity and conduct and utilise this information for various purposes.

In the case of COVID-19, the behaviour data gathered through an IoB app would enable the monitoring of a single individual's or a community's current mass activities. In order to map continuing behaviours onto the context or domain of the behaviour, this can then be complemented by pertinent context data such as geographic, organisational, process, community, medical, economic, or any other background information. The collected data is once more examined using behavioural science. IoB can affect customers' purchasing decisions, lifestyle selections, and other decisions by observing their usage patterns. The IoB improves IoT applications as IoT merely uses the data to act, whereas IoB offers a choice to customers who are most likely to take it. Healthcare professionals have several chances to manage patients utilising particular applications through IoT and IoB. IoT devices may gather metrics like heart rate, blood pressure, temperature, and more and deliver these data to software programmes, enabling remote patient monitoring. Based on the information from IoT devices, IoB can alert users to potential health issues or remind them to take their medications. People will experience efficiency, comfort, and safety in their daily lives with the help of IoB. With the aid of IoB, we can make smarter decisions. During COVID-19, this technology was employed to determine whether or not a person had worn a face mask and washed his hands. They were constantly reminded to abide by the rules to protect themselves.

9. Limitations

IoB can improve our lives significantly, but it also has significant drawbacks, with cybersecurity being the major. Cybercriminals may gain access to behavioural information about consumer buying habits or preferences and their banking information, enabling them to develop sophisticated schemes and elevate phishing to a new level. Despite the concerns mentioned above, IoB can simplify our lives by enhancing business, motivating us to lead healthier lives, or ensuring our safety in the event of pandemics. IoB has already begun to transform the customer experience industry. The difficulty of configuring access levels to users across distributed IoT networks raises concerns about data privacy and integrity. A hacker could reveal sensitive information if they obtain behaviour data. As a result, while implementing IoB-based solutions, businesses should emphasise data integrity and security the most.

On the negative side, IoB is susceptible to online dangers such as unauthorised access to private information that reveals purchasing patterns. Sensitive information, like delivery routes, property access codes, and even bank login codes, might get into the wrong hands and result in irreparable harm. Businesses must be careful and proactive within their area and adhere to stringent data safety requirements as we continue to develop stronger/stricter privacy and data usage, cybersecurity protocols, and regulations to protect us from intrusive data collecting. They must improve their current IT systems and invest in cybersecurity education and awareness campaigns to stay current. The IoB solutions may unintentionally monitor other individuals around the user, violating their privacy, when used in public spaces like schools and hospitals. People with lower incomes and limited access to technology may miss out on IoB benefits as more healthcare professionals and insurance firms incorporate wearable data into treatment plans and health coverage.

10. Future scope of IoB in healthcare

In the future, sales and shop floor employees will be monitored to gauge performance. IoB will therefore have a significant impact on raising industry productivity. In order to extract information from the data gathered by all of these devices and infer behaviours and decision-making tendencies, the IoB intends to transform it. By fusing analytics and behavioural science with data gathered from human behaviour, IoB is taking data processing to a new level. This behavioural data will be crucial in helping businesses plan and create strategies, especially for sales and marketing. It can analyse consumer data and use it for advertising products more effectively and enhancing the overall usability of a product or service, thereby achieving its primary objective of selling products. The IoB principle will become helpful for wearable technologies. Fitbits and smartwatches are examples of wearable technology that can collect data on a user's health and fitness and transmit that data in real-time to a healthcare practitioner.

For instance, corporations, yoga, and many other activities all make extensive use of technology. Any conversation regarding IoB must include a mention of IoT. An online network of physically linked things gathers and shares data and information. The IoB will become more sophisticated due to the way that devices are linked, the calculations they can do independently, and the data stored in the cloud. Businesses can use cutting-edge techniques for marketing goods and services and influencing consumer and employee behaviour by using IoB. Due to the ability to optimise consumer connections based on collected data, this technology will be beneficial to businesses.

11. Conclusion

IoB combines technologies that have been used individually for a long time: extensive data analysis, facial recognition, and location analysis. The IoB, in contrast to the IoT, which connects every component in an environment, essentially builds a global network of live beings. Platforms for customer relationship management will incorporate IoB. In healthcare, implantable and embedded IoB items can alter and repair the body's functions that have been impacted by physical trauma or disease, in contrast to wearables, which only collect data. These include automated insulin administration systems that track blood sugar levels in real-time, connected pacemakers that send data to a specific smartphone app and microelectronic retina prostheses that restore partial vision to patients with retinal disorders. Healthcare professionals may utilise customer data to assess whether people buy junk food at a much higher rate than usual. Based on this behavioural knowledge, healthcare practitioners may engage with particular people to ensure they are not endangering their health. Smart gadgets or applications may be used as health advisors to draw attention to specific health conditions. For instance, athletes may use fitness trackers or other smart equipment to assess their heart rate, daily step count, and calories. IoB products come in a variety of shapes and levels of complexity, from smartwatches and fitness trackers to implantable insulin delivery systems, ingestible sensors, and devices for brain stimulation. Better health condition diagnosis and treatment, individualised insurance

plans, more productivity, and increased public safety are just a few advantages of implementing IoB solutions at scale. IoB solutions can see suspicious activity and sound the alarm before anything happens. It will be beneficial when these start operating more independently because it will make the environment safer for everyone. The best use cases for IoB will undoubtedly be found in any company utilising IoT technology. When users depart from their ideal behaviours, they can support them by helping them make genuine lifestyle changes. The gadgets may easily adjust to the user's behaviours and help them manage their daily activities.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] R. Saravanakumar, P. Bedi, O. Hemakesavulu, N. Thangadurai, E. Poornima, L. Thangavelu, D. Jayadevappa, IoB: Sensors for wearable monitoring and enhancing health care systems, IEEE Instrum. Meas. Mag. 25 (3) (2022) 63–70.

[2] H. Elayan, M. Aloqaily, F. Karray, M. Guizani, Internet of behavior (IoB) and explainable ai systems for influencing IoB behavior, IEEE Netw. (2022).

[3] A. Tsitsika, M. Janikian, T.M. Schoenmakers, E.C. Tzavela, K. Olafsson, S. Wójcik ..., C. Richardson, Internet addictive behavior in adolescence: a cross-sectional study in seven European countries, Cyberpsychol. Behav. Soc. Netw. 17 (8) (2014) 528–535.

[4] P.V. Pushpa, R. Riyaz, Internet of things based context aware remote health care services, in: 2018 IEEE Symposium Series on Computational Intelligence, SSCI, IEEE, 2018, pp. 799–806.

[5] B. Ikharo, A. Obiagwu, C. Obasi, S.U. Hussein, P. Akah, Security for internet-of-things enabled E-health using blockchain and artificial intelligence: A novel integration framework, in: 2021 1st International Conference on Multidisciplinary Engineering and Applied Science, ICMEAS, IEEE, 2021, pp. 1–4.

[6] L.S.M. Whang, S. Lee, G. Chang, Internet over users psychological profiles: a behavior sampling analysis on internet addiction, Cyberpsychol. Behav. 6 (2) (2003) 143–150.

[7] E. Johannessen, A. Henriksen, G. Hartvigsen, A. Horsch, E. Årsand, J. Johansson, Ubiquitous digital health-related data: clarification of concepts, in: Scandinavian Conference on Health Informatics, 2022, pp. 52–58.

[8] M. Al-Shabi, A. Abuhamdah, Using deep learning to detect abnormal behavior in the internet of things, Int. J. Electr. Comput. Eng. 12 (2) (2022) 2108.

[9] P. Adamczewski, The top ICT-trends to accelerate digital transformation in VUCA-environment, IT Pract. 2020 (5) (2020).

[10] Y. Soni, G.C. Gandhi, D. Goyal, A critical analysis on applications and impact of emerging technologies in employment and skills domain in e-governance projects, in: Proceedings of the Third International Conference on Information Management and Machine Intelligence, Springer, Singapore, 2023, pp. 597–606.

[11] M.W. Davis, M.J. Kirwan, W.N. Maclay, H.P. Pappas (Eds.), Closing the Care Gap with Wearable Devices: Innovating Healthcare with Wearable Patient Monitoring, CRC Press, 2022.

[12] L. Aguiar-Castillo, V. Guerra, J. Rufo, J. Rabadan, R. Perez-Jimenez, Survey on optical wireless communications-based services applied to the tourism industry: Potentials and challenges, Sensors 21 (18) (2021) 6282.

[13] A. Celik, A.M. Eltawil, Enabling the internet of bodies through capacitive body channel access schemes, IEEE Internet Things J. (2022).

[14] O. Vermesan, M. Eisenhauer, H. Sundmaeker, P. Guillemin, M. Serrano, E.Z. Tragos ., R. Bahr, Internet of things cognitive transformation technology research trends and applications, Cogn. Hyperconnected Digit. Transform. 1 (2022) 7–95.

[15] A. Heidari, M.A. Jabraeil Jamali, N. Jafari Navimipour, S. Akbarpour, Deep Q-learning technique for offloading offline/online computation in blockchain-enabled green IoT-edge scenarios, Appl. Sci. 12 (16) (2022) 8232.

[16] B. Suruliraj, K. Bessenyei, A. Bagnell, P. McGrath, L. Wozney, R. Orji, S. Meier, Mobile sensing apps and self-management of mental health during the COVID-19 pandemic: Web-based survey, JMIR Form. Res. 5 (4) (2021) e24180.

[17] Antecedents of intention to adopt mobile health (mhealth) application and its impact on intention to recommend: An evidence from Indonesian customers, Int. J. Telemed. Appl. (2021) (2021).

[18] M. Javaid, A. Haleem, R.P. Singh, S. Rab, R. Suman, Internet of behaviours (IoB) and its role in customer services, Sens. Int. (2021) 100122, p. 2.

[19] K.A. Clauson, R.D. Crouch, E.A. Breeden, N. Salata, Blockchain in pharmaceutical research and the pharmaceutical value chain, in: Blockchain in Life Sciences, Springer, Singapore, 2022, pp. 25–52.

[20] K. Sanchez, B. da Graca, L.R. Hall, M.M. Bennett, M.B. Powers, A.M. Warren, The pandemic experience for people with depressive symptoms: Substance use, finances, access to treatment, and trusted sources of information, Subst. Abuse: Res. Treat. (2022) 11782218221126973, p. 16.

[21] T. Rahaman, Smart things are getting smarter: An introduction to the internet of behavior, Med. Ref. Serv. Q. 41 (1) (2022) 110–116.

[22] T. Mezair, Y. Djenouri, A. Belhadi, G. Srivastava, J.C.W. Lin*, Towards an advanced deep learning for the internet of behaviors: Application to connected vehicle, ACM Trans. Sensor Netw. (2022).

[23] F. Barachini, C. Stary, Beyond data: Unifying behavior modeling, in: From Digital Twins to Digital Selves and beyond, Springer, Cham, 2022, pp. 21–33.

[24] S. Stupar, M.B. Ćar, Benefits and risks of applying internet of bodies technology (IoB), in: International Conference New Technologies, Development and Applications, Springer, Cham, 2022, pp. 969–980.

[25] O.H. Embarak, Internet of behaviour (IoB)-based AI models for personalised smart education systems, Procedia Comput. Sci. 203 (2022) 103–110.

[26] O. Embarak, An adaptive paradigm for smart education systems in smart cities using the internet of behaviour (IoB) and explainable artificial intelligence (XAI), in: 2022 8th International Conference on Information Technology Trends, ITT, IEEE, 2022, pp. 74–79.

[27] J. Bzai, F. Alam, A. Dhafer, M. Bojović, S.M. Altowaijri, I.K. Niazi, R. Mehmood, Machine learning-enabled internet of things (IoT): Data, applications, and industry perspective, Electronics 11 (17) (2022) 2676.

[28] N. Mäkitalo, D. Flores-Martin, J. Berrocal, J. Garcia-Alonso, P. Ihantola, A. Ometov ., T. Mikkonen, The internet of bodies needs a human data model, IEEE Internet Comput. 24 (5) (2020) 28–37.

[29] N.U. Sama, K. Zen, M. Humayun, N.Z. Jhanjhi, A.U. Rahman, Security in wireless body sensor network: A multivocal literature study, Appl. Syst. Innov. 5 (4) (2022) 79.

[30] G. Cappon, G. Acciaroli, M. Vettoretti, A. Facchinetti, G. Sparacino, Wearable continuous glucose monitoring sensors: a revolution in diabetes treatment, Electronics 6 (3) (2017) 65.

[31] R. Matsuoka, H. Akazawa, S. Kodera, I. Komuro, The dawning of the digital era in the management of hypertension, Hypertens. Res. 43 (11) (2020) 1135–1140.

[32] S. Mwije, N. Holvoet, Interventions for improving male involvement in maternal and child healthcare in Uganda: A realist synthesis, Afr. J. Reprod. Health 25 (1) (2021) 138–160.

[33] T. Brunschwiler, J. Weiss, S. Paredes, A. Sridhar, U. Pluntke, S.M. Chau ., T. van Kessel, Internet of the body-wearable monitoring and coaching, in: 2019 Global IoT Summit (GIoTS), IEEE, 2019, pp. 1–6.

[34] S. Lewis-Jackson, E. Iob, V. Giunchiglia, J.R. Cabral, M. Romeu-Labayen, S. Cooper ., S. Staudacher, Policies and politics: An analysis of public policies aimed at the reorganisation of healthcare delivery during the COVID-19 pandemic, in: Caring on the Frontline During COVID-19, Palgrave Macmillan, Singapore, 2022, pp. 39–64.

[35] E.E. Lee, J. Torous, M. De Choudhury, C.A. Depp, S.A. Graham, H.C. Kim ., D.V. Jeste, Artificial intelligence for mental health care: clinical applications, barriers, facilitators, and artificial wisdom, Biol. Psychiatry: Cogn. Neurosci. Neuroimaging 6 (9) (2021) 856–864.

[36] A. Haleem, M. Javaid, R.P. Singh, R. Suman, Medical 4.0 technologies for healthcare: Features, capabilities, and applications, Internet Things Cyber-Phys. Syst. (2022).

[37] P.V. Pushpa, Context information modelling for internet of things, in: 2016 2nd International Conference on Contemporary Computing and Informatics (IC3I), IEEE, 2016, pp. 393–399.

[38] F. Ahamad, M.Z. Khan, N. Akhtar, An empirical study on the current state of internet of multimedia things (IoMT), Int. J. Eng. Res. Comput. Sci. Eng. (2021).

[39] P.V. Astillo, G. Choudhary, D.G. Duguma, J. Kim, I. You, TrMAps: trust management in specification-based misbehavior detection system for IMD-enabled artificial pancreas system, IEEE J. Biomed. Health Inf. 25 (10) (2021) 3763–3775.

[40] L.B. Aknin, J.E. De Neve, E.W. Dunn, D.E. Fancourt, E. Goldberg, J.F. Helliwell ., Y. Ben Amor, Mental health during the first year of the COVID-19 pandemic: A review and recommendations for moving forward, Perspect. Psychol. Sci. 17 (4) (2022) 915–936.

[41] A. Heidari, N. Jafari Navimipour, M. Unal, S. Toumaj, Machine learning applications for COVID-19 outbreak management, Neural Comput. Appl. (2022) 1–36.

[42] E. Baldwin, J.R.R. Plomin, A. Steptoe, Adverse childhood experiences, daytime salivary cortisol, and depressive symptoms in early adulthood: a longitudinal genetically informed twin study, Transl. Psychiatry 11 (1) (2021) 1–10.

[43] H.D. Mohammadian, V. Wittberg, M. Castro, G. Bolandian, The 5th wave and i-sustainability plus theories as solutions for SocioEdu consequences of Covid-19, in: LWMOOCS, 2020, pp. 118–123.

[44] S. Fong, C. Bhatt, D. Korzun, S.H. Yang, L. Yang, Internet of breath (IoB): Integrative indoor gas sensor applications for emergency control and occupancy detection, in: First International Conference on Real-Time Intelligent Systems, Springer, Cham, 2017, pp. 342–359.

[45] A.M. Matwyshyn, The internet of bodies, Wm. Mary L. Rev. 61 (77) (2019).

[46] C.M. Charron, K.M. Gorey, Virtual versus face-to-face cognitive behavioral treatment of depression: Meta-analytic test of a noninferiority hypothesis and men's mental health inequities, Depress. Res. Treat. (2021) 2022.

[47] C. Amato, Internet of bodies: Digital content directive, and beyond, J. Intell. Prop. Inf. Tech. Electr. Com. L 12 (181) (2021).

[48] Y.B. Zikria, S.W. Kim, O. Hahm, M.K. Afzal, M.Y. Aalsalem, Internet of things (IoT) operating systems management: Opportunities, challenges, and solution, Sensors 19 (8) (2019) 1793.

[49] C. West, C. Rouen, Covid-19: Implications for mental health and well-being, now and in the digital future, in: Digital Transformation in a Post-COVID World, CRC Press, 2021, pp. 3–22.

[50] L. Vuillier, L. May, M. Greville-Harris, R. Surman, R.L. Moseley, The impact of the COVID-19 pandemic on individuals with eating disorders: the role of emotion regulation and exploration of online treatment experiences, J. Eating Disorders 9 (1) (2021) 1–18.

[51] A. Hampshire, P.J. Hellyer, E. Soreq, M.A. Mehta, K. Ioannidis, W. Trender ., S.R. Chamberlain, Associations between dimensions of behaviour, personality traits, and mental health during the COVID-19 pandemic in the United Kingdom, Nature Commun. 12 (1) (2021) 1–15.

[52] C. Stary, Digital twin generation: Re-conceptualising agent systems for behavior-centered cyber–physical system development, Sensors 21 (4) (2021) 1096.

[53] Q. Wang, Opportunities and challenges faced by IoB in digital medical and health communication, Policy 6 (2) (2022) 57–63.

[54] G. Gustin, B. Macq, D. Gruson, S. Kieffer, Empowerment of diabetic patients through mhealth technologies and education: development of a pilot self-management application, in: 13th International Conference on Medical Information Processing and Analysis, Vol. 10572, SPIE, 2017, pp. 167–177.

[55] A. Minutolo, E. Damiano, G. De Pietro, H. Fujita, M. Esposito, A conversational agent for querying Italian patient information leaflets and improving health literacy, Comput. Biol. Med. 141 (2022) 105004.

[56] C.D. Bergeron, A. Boolani, E.C. Jansen, M.L. Smith, Practical solutions to address COVID-19-related mental and physical health challenges among low-income older adults, Front. Publ. Health (2021) 929.

[57] D.P. Möller, Guide to computing fundamentals in cyber–physical systems, in: Computer Communications and Networks, Springer, Heidelberg, 2016.

[58] A. Majeed, S. Afzal, M. Amer, A narrative review of mental health landscape of survivors, healthcare workers, and general public in the post-COVID world, Anaesthesia Pain Intensive Care 25 (4) (2021) 513–518.

[59] Y. Cao, J. Zhang, L. Ma, X. Qin, J. Li, Examining users' initial trust building in mobile online health community adopting, Int. J. Environ. Res. Public Health 17 (11) (2020) 3945.

[60] E. Iob, A. Steptoe, P. Zaninotto, Mental health, financial, and social outcomes among older adults with probable COVID-19 infection: A longitudinal cohort study, Proc. Natl. Acad. Sci. 119 (27) (2022) e2200816119.

[61] A. Giordanengo, E. Årsand, A.Z. Woldaregay, M. Bradway, A. Grottland, G. Hartvigsen ., A.H. Hansen, Design and prestudy assessment of a dashboard for presenting self-collected health data of patients with diabetes to clinicians: Iterative approach and qualitative case study, JMIR Diabetes 4 (3) (2019) e14002.

[62] L. Meneghetti, M. Terzi, S. Del Favero, G.A. Susto, C. Cobelli, Data-driven anomaly recognition for unsupervised model-free fault detection in artificial pancreas, IEEE Trans. Control Syst. Technol. 28 (1) (2018) 33–47.

[63] P. Nagaraj, P. Deepalakshmi, M.F. Ijaz, Optimised adaptive tree seed Kalman filter for a diabetes recommendation system—bilevel performance improvement strategy for healthcare applications, in: Cognitive and Soft Computing Techniques for the Analysis of Healthcare Data, Academic Press, 2022, pp. 191–202.

[64] J. Lindert, M. Jakubauskiene, J. Bilsen, The COVID-19 disaster and mental health—assessing, responding and recovering, Eur. J. Publ. Health 31 (Supplement_4) (2021) iv31–iv35.

[65] C. Stary, The internet-of-behavior as organisational transformation space with choreographic intelligence, in: International Conference on Subject-Oriented Business Process Management, Springer, Cham, 2020, pp. 113–132.

[66] X. Yuan, H. Tian, Z. Zhang, Z. Zhao, L. Liu, A.K. Sangaiah, K. Yu, A MEC offloading strategy based on improved DQN and simulated annealing for internet of behavior, ACM Trans. Sensor Netw. (2022).

[67] A. Halgekar, A. Chouhan, I. Khetan, J. Bhatia, N. Shah, K. Srivastava, Internet of behavior (IoB): A survey, in: 2021 5th International Conference on Information Systems and Computer Networks, ISCON, IEEE, 2021, pp. 1–6.

[68] M. Mariello, K. Kim, K. Wu, S.P. Lacour, Y. Leterrier, Recent advances in encapsulation of flexible bioelectronic implants: materials, technologies and characterisation methods, Adv. Mater. (2022) 2201129.

[69] P. Shah, J. Hardy, M. Birken, U. Foye, R. Rowan Olive, P. Nyikavaranda ., B. Lloyd-Evans, What has changed in the experiences of people with mental health problems during the COVID-19 pandemic: a coproduced, qualitative interview study, Soc. Psychiatry Psychiatr. Epidemiol. 57 (6) (2022) 1291–1303.

[70] V.M. Constantino, B.M. Fregonesi, K.A.D.A. Tonani, G.S. Zagui, A.P.C. Toninato, E.R.D.S. Nonose ., S.I. Segura-Muñoz, Storage and disposal of pharmaceuticals at home: a systematic review, Ciencia Saude Coletiva 25 (2020) 585–594.

[71] E. Robinson, M. Daly, Explaining the rise and fall of psychological distress during the COVID-19 crisis in the United States: Longitudinal evidence from the understanding America study, Br. J. Health Psychol. 26 (2) (2021) 570–587.

[72] J. Liu, D.J. Spakowicz, G.I. Ash, R. Hoyd, R. Ahluwalia, A. Zhang ., M. Gerstein, Bayesian structural time series for biomedical sensor data: A flexible modeling framework for evaluating interventions, PLoS Comput. Biol. 17 (8) (2021) e1009303.

[73] M. Daly, E. Robinson, Acute and longer-term psychological distress associated with testing positive for COVID-19: longitudinal evidence from a population-based study of US adults, Psychol. Med. (2021) 1–8.

[74] R.R. Tambling, B.S. Russell, M. Fendrich, C.L. Park, Predictors of mental health help-seeking during COVID-19: Social support, emotion regulation, and mental health symptoms, J. Behav. Health Serv. Res. (2022) 1–12.

[75] S. Wang, Y. Hou, F. Gao, X. Ji, A reconfigurable smart interface based on IEEE 1451 and field programmable gate array for multiple internet of things devices, Int. J. Distrib. Sens. Netw. 13 (2) (2017) 1550147717693848.

[76] N.S. Gluckman, A. Eagle, M. Michalitsi, N. Reynolds, Adapting to the COVID-19 pandemic: A psychological crisis support call service within a community mental health team, Community Ment. Health J. (2022) 1–10.

[77] M. Rahman, R. Ahmed, M. Moitra, L. Damschroder, R. Brownson, B. Chorpita ., M. Kumar, Mental distress and human rights violations during COVID-19: a rapid review of the evidence informing rights, mental health needs, and public policy around vulnerable populations, Front. Psychiatry 11 (2021) 603875.

[78] A. Kataria, D. Agrawal, S. Rani, V. Karar, M. Chauhan, Prediction of blood screening parameters for preliminary analysis using neural networks, in: Predictive Modeling in Biomedical Data Mining and Analysis, Academic Press, 2022, pp. 157–169.

[79] A. Coravos, J.C. Goldsack, D.R. Karlin, C. Nebeker, E. Perakslis, N. Zimmerman, M.K. Erb, Digital medicine: a primer on measurement, Digit. Biomark. 3 (2) (2019) 31–71.

[80] D. Singh, A.K. Maurya, R.K. Dewang, N. Keshari, A review on internet of multimedia things (IoMT) routing protocols and quality of service, Internet Multimedia Things (IoMT) (2022) 1–29.

[81] T.S. Bailey, J.Y. Stone, A novel pen-based bluetooth-enabled insulin delivery system with insulin dose tracking and advice, Expert Opin. Drug Deliv. 14 (5) (2017) 697–703.

[82] O. Flygare, E. Andersson, G. Glimsdal, D. Mataix-Cols, D. Pascal, C. Rück, J. Enander, Cost-effectiveness of internet-delivered cognitive behaviour therapy for body dysmorphic disorder: results from a randomised controlled trial, Internet Interventions (2023) 100604.

[83] M. Javaid, S. Khan, A. Haleem, S. Rab, Adoption of modern technologies for implementing industry 4.0: an integrated MCDM approach, Benchmark.: Int. J. (2022).

[84] M.M. Islam, S. Nooruddin, F. Karray, G. Muhammad, Multi-level feature fusion for multimodal human activity recognition in internet of healthcare things, Inf. Fusion (2023).

[85] M. Javaid, A. Haleem, R.P. Singh, S. Khan, Understanding roles of virtual reality in radiology, Internet Things Cyber-Phys. Syst. 2 (2022) 91–98.

[86] A. Haleem, M. Javaid, R.P. Singh, R. Suman, S. Khan, Management 4.0: Concept, applications and advancements, Sustain. Oper. Comput. 4 (2023) 10–21.

[87] S.J. Melhem, S. Nabhani-Gebara, R. Kayyali, Digital trends, digital literacy, and e-health engagement predictors of breast and colorectal cancer survivors: a population-based cross-sectional survey, Int. J. Environ. Res. Public Health 20 (2) (2023) 1472.

[88] S. Khan, R. Singh, J.C. Sá, G. Santos, L.P. Ferreira, Modelling of determinants of logistics 4.0 adoption: Insights from developing countries, Machines 10 (12) (2022) 1242.

[89] H.P. Sharma, A. Chaturvedi, Adoption of smart technologies: An Indian perspective, in: 2021 5th International Conference on Information Systems and Computer Networks, ISCON, IEEE, 2021, pp. 1–4.

[90] R. Singh, S. Khan, J. Dsilva, P. Centobelli, Blockchain integrated IoT for food supply chain: A grey based delphi-DEMATEL approach, Appl. Sci. 13 (2) (2023) 1079.

[91] P. Paul, B. Singh, Healthcare employee engagement using the internet of things: A systematic overview, Adopt. Effect Artif. Intell. Human Resour. Manage. A 7 (2023) 1–97.

[92] S. Akkol-Solakoglu, D. Hevey, Internet-delivered cognitive behavioural therapy for depression and anxiety in breast cancer survivors: Results from a randomised controlled trial, Psycho-Oncol. (2023).

[93] M. Javaid, A. Haleem, R.P. Singh, R. Suman, S. Khan, A review of blockchain technology applications for financial services, BenchCouncil Trans. Benchmark. Standards Eval. (2022) 100073.

[94] E. Liberati, N. Richards, J. Parker, J. Willars, D. Scott, N. Boydell ., P. Jones, Remote care for mental health: qualitative study with service users, carers and staff during the COVID-19 pandemic, BMJ Open 11 (4) (2021) e049210.

[95] I.H. Khan, M.I. Khan, S. Khan, Challenges of IoT implementation in smart city development, in: Smart Cities—Opportunities and Challenges: Select Proceedings of ICSC 2019, Springer Singapore, Singapore, 2020, pp. 475–486.

[96] M.I. Khan, S. Khan, U. Khan, A. Haleem, Modeling the big data challenges in context of smart cities–an integrated fuzzy ISM-DEMATEL approach, Int. J. Build. Pathol. Adapt. (2021).

[97] C. Martín, J. Hoebeke, J. Rossey, M. Díaz, B. Rubio, F. Van den Abeele, Adaptivity: An internet of things device-decoupled system for portable applications in changing contexts, Sensors 18 (5) (2018) 1345.

[98] T.S. Jesus, S. Bhattacharjya, C. Papadimitriou, Y. Bogdanova, J. Bentley, J.C. Arango-Lasprilla ., Refugee Empowerment Task Force, International Networking Group of the American Congress of Rehabilitation Medicine, Lockdown-related disparities experienced by people with disabilities during the first wave of the COVID-19 pandemic: Scoping review with thematic analysis, Int. J. Environ. Res. Public Health 18 (12) (2021) 6178.

BenchCouncil Transactions on Benchmarks, Standards and Evaluations 2 (2022) 100082

Contents lists available at ScienceDirect

BenchCouncil Transactions on Benchmarks, Standards and Evaluations

journal homepage: www.keaipublishing.com/en/journals/benchcouncil-transactions-on-benchmarks-standards-and-evaluations/

Short Communication

Enabling Reduced Simpoint Size Through LiveCache and Detail Warmup

Jose Renau [a], Fangping Liu [b], Hongzhang Shan [b], Sang Wook Stephen Do [b,*]

[a] Uncore LLC, Santa Cruz, CA, US
[b] IC LAB, Futurewei Technologies, Santa Clara, CA, US

ARTICLE INFO

Keywords:
Architecture
Simpoint
LiveCache
Detail-warmup
Simulation sampling
Cycle-accurate simulation
Microarchitecture simulation

ABSTRACT

Simpoint technology (Sherwood et al., 2002) has been widely used by modern micro-architecture research community to significantly speedup the simulation time. However, the typical Simpoint size remains to be tens to hundreds of million instructions. At such sizes, the cycle-accurate simulators still need to run tens of hours or even days to finish the simulation, depending on the architecture complexity and workload characteristics. In this paper, we developed a new simulation framework by integrating LiveCache and Detail-warmups with Dromajo (https://chipyard.readthedocs.io/en/latest/Tools/Dromajo.html) and Kabylkas et al. (2005), enabling us to use much smaller Simpoint size (2 million instructions) without loss of accuracy. Our evaluation results showed that the average simulation time can be accelerated by 9.56 times over 50M size and most of the workload simulations can be finished in tens of minutes instead of hours.

1. Introduction

Modern computer architecture researches rely heavily on computer simulation to study new architectural features or estimate the performance, power, and area. A cycle-accurate simulator often takes hundreds of simulation hours, prohibiting its use in practice. To expedite the simulation, prior arts have explored various techniques. Sampling is one of the popular approaches, where a simulator runs sampled executions instead of the entire benchmark. The sampling could be based on either statistical sampling [1–3] or representative sampling [4].

Simpoint [4,5] is one of the most widely used sampling techniques, which could reduce the simulation time dramatically from months to days to hours. Running only the representative checkpoints of a program execution so-called 'Simpoints' generated by the Simpoint toolset enables computer architecture simulation to finish earlier than running the same program from the beginning to the end. However, the typical Simpoint size remains to be tens to hundreds of million instructions to maintain the accuracy. Depending on the architectural complexity, it still takes tens of hours to finish, still not fast enough for a quick turnaround.

In this paper, we developed a framework based on Dromajo [6,7] to enable us to use Simpoints with only 2 million instructions (2M). Compared with regular Simpoints with hundreds of million instructions or over, the simulation time could be greatly reduced from hours to minutes without loss of accuracy. Dromajo is a RISC-V RV64GC emulator, which enables executing an application under fast software simulation, generating checkpoints after a given number of instructions, and resuming such checkpoints in another slow, cycle-accurate simulator to generate micro-architecture simulation results.

In order to use smaller Simpoint size, one challenge needs to be addressed is the simulator needs to start from the up to date architectural status. Otherwise the simulation accuracy cannot be maintained. Large simpoint sizes may obviate this need. To fulfill this purpose, we integrate the LiveCache technique from [8,9] into Dromajo so that Dromajo can record the memory operations in timing order up to the Simpoint location in the checkpoint files. The number of memory operations to be recorded is a configuration parameter set accordingly with the cache size(s) of the simulated target micro-architecture. When the cycle-accurate simulator starts, by reading the checkpoint files, it can repeat these memory operations and bring the cache status up to date quickly. To bring the status of other architectural components up to date, such as a branch predictor, we resort to Detail-warmup, which allows us to run a specified number of instructions right before the Simpoint location, from which the simulator is dictated to start to measure the performance numbers. Correspondingly, the actual Simpoint locations will also be adjusted based on the number of instructions defined by Detail-warmup.

Putting all together, Simpoint execution is preceded by LiveCache-warmup first, followed by the Detail-warmup, then starts to collect the performance numbers thereafter. Compared with running only Simpoint itself, LiveCache and Detail-warmup enable us to bring the machine status up to date, preserving the simulation accuracy. As far as we know, this is the first framework combining both LiveCache and Detail-warmup together to generate 2 million Simpoints on RISC-V platforms.

* Corresponding author.
E-mail addresses: renau@uncore.io (J. Renau), julius.liu@futurewei.com (F. Liu), hshan@futurewei.com (H. Shan), sdo@futurewei.com (S.W.S. Do).

https://doi.org/10.1016/j.tbench.2022.100082
Received 23 September 2022; Received in revised form 12 December 2022; Accepted 12 December 2022
Available online 21 December 2022

In summary, we contribute the followings to the state of the art:

(1) Developed an open source framework that enables us to make use of smaller (2M) Simpoint size without loss of accuracy.
(2) Evaluated the 2M Simpoint size with SPEC 2006 CPU benchmark suite. Compared with 50M Simpoint size, the average simulation time has been improved more than 9 times.
(3) Quantitatively study the performance effects of LiveCache and Detail-warmup on simulation speed and accuracy.

Including the LiveCache technique, prior works such as [1–3,8,10–21] have proposed and discussed various micro-architecture warm-up techniques and effects, to which we plan to extend our work.

The rest of the paper is divided into the following sections. Section 2 describes the implementation. Section 3 presents the evaluation results with analysis. Lastly, Section 4 concludes the paper with comments on future directions.

2. Implementation

Our implementation is based on Dromajo [6,7], an open source RISC-V emulator. We extended Dromajo so that it is capable of generating Simpoints with user-configurable LiveCache and Detail-warmup.

2.1. Base Simpoint creation and execution

First, we modified the Dromajo source code to enable it to profile a benchmark based on Simpoint requirements. The profiling should follow the basic-block characterization described in the Simpoint papers [4,5]. We used the Dromajo's existing checkpointing option to generate two checkpoint files at each Simpoint location. The first file includes RISC-V instructions to be executed to restore the target machine's architectural state such as the contents of the physical register file and the control registers at the time of the corresponding Simpoint creation. The second file contains the memory image including the instruction and data areas with others necessary memory contents to run the benchmark. As far as we know, we have first used, designed and implemented Simpoint support on Dromajo since it was first discussed in [7].

To run Simpoints, we modified our target cycle-accurate simulator to copy the memory image into the target simulator's memory space and let the execution start from the first instruction in the first Simpoint (checkpoint) file. The RISC-V 'dret' instruction inserted by the Dromajo checkpointing option at the end should have execution jump to the desired Simpoint location. We also modified the target simulator to reset and start to (re)collect simulation statistics such as the number of cache misses and branch mispredictions, etc. right after the dret instruction execution.

2.2. LiveCache

For LiveCache, we implement similar mechanism described in [9], which adopts the MTR (Memory Timestamp Record) technique from [8], on top of the base Simpoint framework as described above to have Dromajo record memory operations up to the current Simpoint location and translate them into RISC-V 'load' instructions for clean cache lines or 'load' and 'store' instruction pairs for dirty cache lines using the memory addresses recorded. These load and store instructions are stored in the first Simpoint file and will be executed later by simulator to bring the cache status up to date. The total simulation time should increase accordingly because of the execution time of these additional load and store instructions.

To limit the number of the LiveCache load and store instructions, our framework takes 'Bootrom' size as an input parameter. For example, if the Bootrom size were 8 KB, then there are 1024 64-bit addresses recorded that corresponds to a maximum of 1024 loads or load and store pairs. The actual Bootrom size should be set based on the cache size(s) of the simulated target micro-architecture.

For the verification, we tested the following C language code snippet on our target cycle-accurate simulator. We made two checkpoints at the third loop iteration with LiveCache on and off. The results showed that with LiveCache on, the IPC was improved about eleven percent, confirming the effectiveness of the mechanism.

```
   //an array of 32k 32-bit words
   //occupying 2048 cache lines
10 int vec[0x8000];
20 register int sum = 0;
30 for (int i = 0; i < 5; ++i) {
40  for (int j = 0; j < 0x8000; j += 16) {
     //do one load action
     //each cacheline or 64 bytes
50   sum += vec[j];
60  }
70 }
```

2.3. Detail-warmup

Detail-warmup aims at warming up micro-architecture components such as branch predictor and instruction and data caches by executing some instructions right before a Simpoint while LiveCache specifically aims at data caches. The goal is to restore the machine state of a target simulator as close as possible as if the target simulator has kept running up to the Simpoint location.

The implementation goal is to make a checkpoint at a location prior to a Simpoint by the number of instructions specified by the Detail-warmup size parameter. The base Simpoint creation does not consider these additional instructions, and we had to add an additional step to adjust the Simpoint location accordingly. We intervened the base Simpoint creation process right before the final step with our scripts to adjust the actual Simpoint location earlier by the Detail-warmup size. For a rare case where a base Simpoint needs to be created at the very beginning of execution, the whole Simpoint window needs to be adjusted to make room for Detail-warmup because a Simpoint location cannot be specified prior to the very beginning.

From the description, we can find that Detail-warmup is more powerful than LiveCache to update the architectural states. However, Detail-warmup is much more expensive. LiveCache can help to reduce the Detail-warmup size. It is the combination of LiveCache and Detail-warmup that enables smaller Simpoint size fast and accurate.

In summary, we applied the LiveCache and Detail-warmup modifications to related places in the Dromajo source code, where the original code adds the LiveCache memory instructions and captures the corresponding architectural snapshot in a checkpoint file respectively. We expect that one can port the modifications in a different tool set other than Dromajo.

3. Evaluation

3.1. Simulation setup

For the benchmark setup, we use the SPEC CPU 2006 benchmark suite [22]. We use Buildroot [23] to include a Linux kernel in Simpoint for the system call support.

For the target simulator setup, we use our in-house cycle-accurate simulator, which runs unmodified RISC-V instructions. The target simulator also implements hardware support to handle interrupts and exceptions based on the RISC-V specification to run the benchmark applications as intended. Table 1 shows the target simulator configuration.

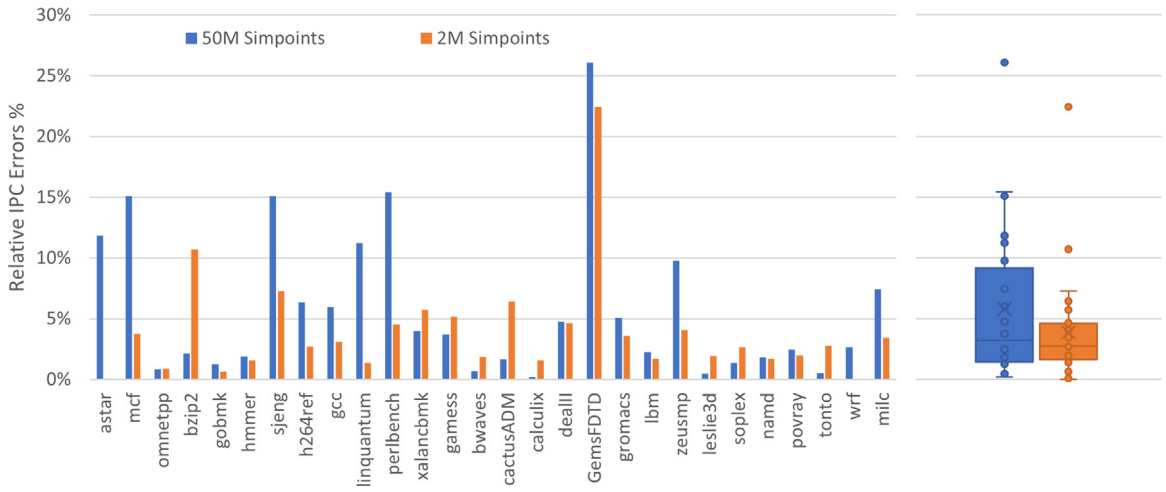


Fig. 1. The relative IPC errors for 2M Simpoints (with LiveCache and 4M Detail-warmup instructions) and 50M Simpoints (with 50M Detail-warmup). Using 2M Simpoints can reduce the IPC errors from average 5.46% to 3.89%.

Table 1
Simulator configuration.

Core	Single-Core, 96 Inst. Q entries dispatch width - 8 Int, 4 Fp instructions 3-ALU, DIV, MUL, FP 48-bit VA, 40-bit MAX PA
Instruction fetch	64 KB, 4-way, 64B-line Fully pipelined 48-entry TLB, 32-entry RAS BTB, TAGE predictor
L1 data	64 KB, 4-way, 64B-line LRU, 4-cycle latency
L2	1 MB, 8-way, 64B-line LRU, 9-cycle latency
L3	4 MB, 16-way, 64B-line LRU, 13-cycle latency
Memory	167-cycle latency

Table 2
Simpoint configuration.

Setting	Description
Execution window size	10B instructions
Simpoint size	2M instructions
Bootrom size	256 KB
LiveCache-warmup	On or Off
Detail-warmup	0, 2, or 4M instructions

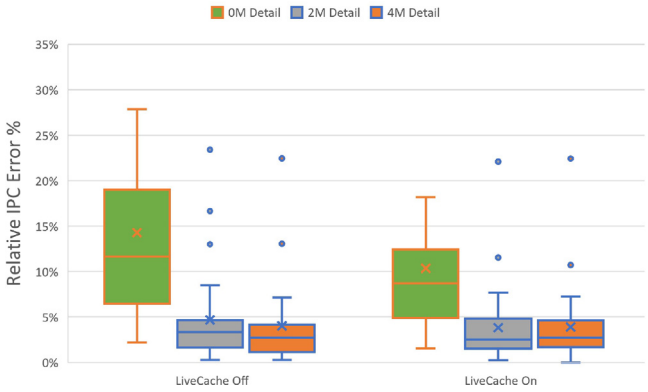


Fig. 2. The relative IPC errors for cases of LiveCache on/off and 0M/2M/4M for Detail-warmup. The best result is obtained with 4M Detail-warmup and LiveCache together while the worst result is from case running 2M Simpoints without LiveCache nor Detail-warmup (LiveCache off/0M Detail-warmup).

For the Simpoint setup, we choose a 10 billion execution window size to avoid very long simulation time, which still allow us to conduct a fair evaluation. We also have Dromajo move this 10 billion instruction window by 100 million instructions to allow Detail-warmup for the case where a Simpoint is created at the very beginning of the execution.

Table 2 shows the Simpoint configuration used for evaluation.

3.2. Simulation results

This section focuses on the evaluation of accuracy and speed of our Simpoint approach. We compare our 2M Simpoint results with the popular 50M Simpoint ones using SPEC CPU2006 benchmark suite as our driving applications, which includes 28 integer and floating-point applications. Sphinx3 is not included currently due to that our RISC-V simulator could not handle its input data set correctly.

3.2.1. IPC accuracy

To compare the accuracy, we compute the IPC (instructions per cycle) errors relative to the 10 billion instruction reference mentioned before. Fig. 1 shows the relative errors of our 2M Simpoints (with LiveCache on and 4M Detail-warmup) and the 50M Simpoints (with 50M Detail-warmup, a similar approach to [24,25]) with both bar chart (left side, for individual results) and whisker box (right side, for overall results) for all 28 individual applications. The whisker boxes illustrate that the 2M Simpoints incur lower mean error (3.89%) than the popular 50M Simpoints (5.80%). Also, the 2M Simpoints have lower maximum error value and tighter bound ranges. The minimum error values are similar, all close to zero. Clearly, 2M Simpoints exhibit a more concentrated error distribution with higher IPC accuracy.

There is one outlier in the whisker box, *GemsFDTD* which solves the Maxwell equations in 3D in the time domain using the finite-difference time-domain method. Neither 2M size nor 50M size works well with this application. Both sizes produce much higher IPC results than the reference. The full IPC trace has a high IPC with short low IPC spikes that Simpoint does not capture correctly. We believe this should be related with the statistical approach used by Simpoint technology. Further study is out of the scope of this short paper.

3.2.2. Detail-warmup and LiveCache effects

The 2M Simpoint results shown in Fig. 1 is obtained with LiveCache and 4M Detail-warmup. To understand their individual effects on the IPC accuracy, we displayed the corresponding results using whisker plot

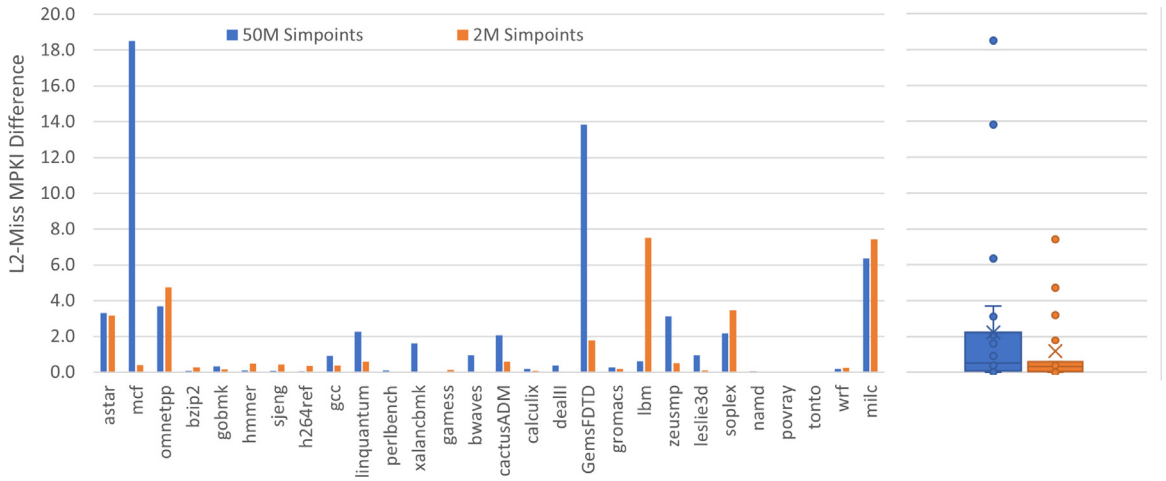


Fig. 3. The differences of L2 misses per thousand instructions (MPKI) with the base reference for both 50M simpoints (with 50M detail warmup) and 2M simpoints (with Livecache and 4M Detail-warmup).

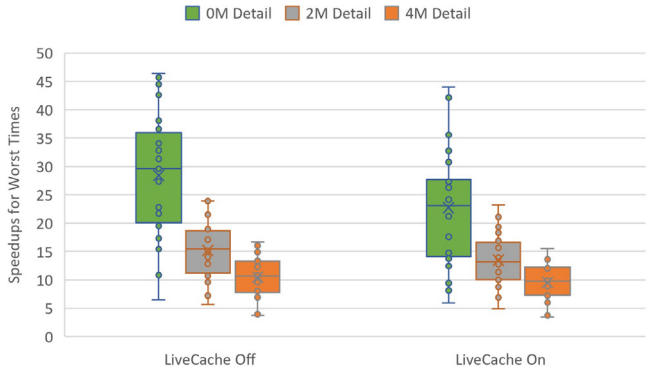


Fig. 4. The worst simulation time speedups of using 2M Simpoint size over 50M Simpoint. The speedup is about 10 times when running 2M Simpoints with LiveCache and 4M Detail-warmup.

in Fig. 2 for six cases: LiveCache on/off, and 0M/2M/4M for Detail-warmup. The leftmost box is for running 2M simpoints without LiveCache nor Detail-warmups (LiveCache off/0M Detail). Clearly, it generates the highest errors. Enabling either LiveCache or Detail-warmups is essential to improve the accuracy.

First, we examine the performance impact of LiveCache. By comparing the cases with LiveCache on and off (left three whisker boxes vs. corresponding right three whisker boxes), we find that, the errors obtained with LiveCache on concentrated on narrower box ranges and all maximum, mean, and median error values are smaller. Such difference is more phenomenal when there is no Detail-warmup (0M instruction case). With the increase of the Detail-warmup size, LiveCache-warmup effect becomes less and less important. However, increasing Detail-warmup size will surely increases the simulation time. Using LiveCache allows us to shorten the Detail-warmup time so that we can avoid the problem of spending a large amount of simulation time on Detail-warmups [1].

Similarly, the IPC accuracy can be significantly improved when increase the Detail-warmup instructions from 0 to 2M. From 2M to 4M, the results can be still be improved. However, using 8M or larger sizes, the accuracy can no longer be further improved. Our best results are obtained when 4M Detail-warmup size are used.

3.2.3. L2 misses

In addition to the IPC accuracy, we also compare the number of L2 misses, one import metric to measure the memory performance, to observe the direct effects on the cache, although those should be reflected in the IPC results. Fig. 3 shows the MPKI (misses per thousand instructions) differences with the 10 billion base reference for both the 50M simpoints with 50M Detail-warmups and 2M simpoints with LiveCache and 4M Detail-warmups. The left bar chart displays the results for individual benchmark (Lower value indicates the MPKI difference with the base reference is smaller) while the right whisker box illustrates the overall results.

Different from Fig. 1, the use of absolute differences instead of relative ratios in Fig. 3 is due to the fact that for some applications, the L2 MPKI is quite small. Using ratios may exaggerate the differences and lead to incorrect conclusions. For example, the L2 MPKI for *tonto* is only 0.03 for its base reference. For 50M simpoints and 2M simpoints, they are 0.01 and 0.03, respectively. Both are very close to the base case. If we use relative errors, the differences between 50M simpoints and 2M simpoints will be 67% ((0.03–0.01)/0.03) and 0% ((0.03–0.03)/0.03), respectively, which does not accurately reflect reality.

In summary, Fig. 3 shows that, similar to the IPC accuracy, using 2M simpoints not only significantly reduces the maximum error but also delivers much higher average accuracy. Also, the 2M simpoint results are obtained with both LiveCache and 4M Detail-warmups. Running only 2M simpoints itself will generate much higher errors and must be accompanied with LiveCache and Detail-warmup to maintain the accuracy.

3.2.4. Simulation speed

Using 2M Simpoint size instead of 50M, we expect the simulation time can be greatly accelerated, ideally 50 times ((50M + 50M)/2M). However, considering the LiveCache and Detail-warmup overhead, especially the Detail-warmup overhead, the actual speedups will be much lower. We compare the running times of 2M and 50M Simpoint sizes using the worst Simpoint simulation time. All the Simpoints of an application are launched at the same time in parallel until all Simpoints are finished. The benchmark running time is determined by the Simpoint with longest simulation time.

Fig. 4 shows the speedups of using 2M Simpoint size over 50M size in terms of worst running times. With LiveCache only without Detail-warmup, the maximum speedup is about 45 times, close to the ideal expectation of 50 times speedup. The average speedup is about 23. Turning the LiveCache on introduces a constant overhead of reading the data file and loading the data into caches. It takes about 2 min with our simulator. However, comparing with Detail-warmup, its effect on the simulation time is relatively small. With the increase of the Detail-warmup size, the average speedups decreases, falling to around 9 for 4M warmup size. The actual average running time for 50M size is about 450 min while for the 2M size, the average real running times are 22, 34, and 49 min for 0M, 2M, and 4M Detail-warmups, respectively.

4. Conclusion

In this paper, we propose a framework to create reduced size Simpoints for simulation sampling, further reducing simulation time of standard Simpoint simulation with slightly improved accuracy. We achieve the goal by incorporating well established LiveCache and Detail-warmup techniques into our base Simpoint framework. The framework shall benefit whoever relies on computer architecture simulation by significantly reducing simulation time with decent sampling error.

Future works include, but not limited to, studying the performance effects of LiveCache and Detail-warmups in detail, extending our framework to support multi-core multi-threaded benchmark applications; exploring and incorporating various other warmup techniques; and enhancing the checkpoint capability to an arbitrary location of interest.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Authors are currently employed by Futurewei Technologies.

References

[1] R.E. Wunderlich, T.F. Wenisch, B. Falsafi, J.C. Hoe, SMARTS: accelerating microarchitecture simulation via rigorous statistical sampling, in: Proceedings of the 30th Annual International Symposium on Computer Architecture, ISCA, 2003, pp. 84–95.

[2] T.M. Conte, M.A. Hirsch, K.N. Menezes, Reducing state loss for effective trace sampling of superscalar processors, in: Proceedings International Conference on Computer Design: VLSI in Computers and Processors, ICCD, 1996, pp. 468–477.

[3] T.M. Conte, M.A. Hirsch, W.W. Hwu, Combining trace sampling with single pass methods for efficient cache simulation, IEEE Trans. Comput. 47 (6) (1998) 714–720.

[4] T. Sherwood, E. Perelman, G. Hamerly, B. Calder, Automatically characterizing large scale program behavior, in: Proceedings of the 10th International Conference on Architectural Support for Programming Languages and Operating Systems, ASPLOS, 2002, pp. 45–57.

[5] E. Perelman, G. Hamerly, B. Calder, Picking statistically valid and early simulation points, in: Proceedings of the 12th International Conference on Parallel Architectures and Cimpilation Techniques, PACT, 2003, pp. 244–255.

[6] https://chipyard.readthedocs.io/en/latest/Tools/Dromajo.html.

[7] N. Kabylkas, T. Thorn, S. Srinath, P. Xekalakis, J. Renau, Effective processor verification with logic fuzzer enhanced co-simulation, in: Proceedings of the 54th International Symposium on Microarchitecture, MICRO, 2005, pp. 667–678.

[8] K.C. Barr, H. Pan, M. Zhang, K. Asanovic, Accelerating multiprocessor simulation with a memory timestamp record, in: Proceedings of 2005 IEEE International Symposium on Performance Analysis of Systems and Software, ISPASS, 2005, pp. 66–77.

[9] S. Hassani, G. Southern, J. Renau, LiveSim: Going live with microarchitecture simulation, in: Proceedings of 2016 IEEE International Symposium on High Performance Computer Architecture, HPCA, 2016, pp. 606–617.

[10] A. Agarwal, J. Hennessy, M. Horowitz, Cache performance of operating system and multiprogramming workloads, ACM Trans. Comput. Syst. 6 (4) (1988) 393–431.

[11] S. Laha, J.A. Patel, R.K. Iyer, Accurate low-cost methods for performance evaluation of cache memory systems, IEEE Trans. Comput. 37 (11) (1988) 1325–1336.

[12] S.K. Reinhardt, M.D. Hill, J.R. Larus, A.R. Lebeck, J.C. Lewis, D.A. Wood, The wisconsin wind tunnel: Virtual prototyping of parallel computers, in: Proceedings of the 1993 ACM SIGMETRICS Conference on Measurement and Modeling of Computer Systems, SIGMETRICS, 1993, pp. 48–60.

[13] R.E. Kessler, M.D. Hill, D.A. Wood, A comparison of trace-sampling techniques for multi-megabyte caches, IEEE Trans. Comput. 43 (6) (1994) 664–675.

[14] J.W. Haskins, K. Skadron, Minimal subset evaluation: Rapid warm-up for simulated hardware state, in: Proceedings of 2001 IEEE International Conference on Computer Design: VLSI in Computers and Processors, ICCD, 2001, pp. 32–39.

[15] J.W. Haskins, K. Skadron, Memory reference reuse latency: Accelerated warmup for sampled microarchitecture simulation, in: Proceedings of 2003 IEEE International Symposium on Performance Analysis of Systems and Software, ISPASS, 2003, pp. 195–203.

[16] Y. Luo, L. John, L. Eeckhout, Self-monitored adaptive cache warm-up for microprocessor simulation, in: Proceedings of 16th Symposium on Computer Architecture and High Performance Computing (SBAC-PAD), 2004, pp. 10–17.

[17] L. Eeckhout, Y. Luo, K.D. Bosschere, L.K. John, BLRL: Accurate and efficient warmup for sampled processor simulation, Comput. J. 48 (4) (2005) 451–459.

[18] T.F. Wenisch, R.E. Wunderlich, B. Falsafi, J.C. Hoe, TurboSMARTS: Accurate microarchitecture simulation sampling in minutes, in: Proceedings of the 2005 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems, 2005, pp. 408–409.

[19] T.F. Wenisch, R.E. Wunderlich, B. Falsafi, J.C. Hoe, Simulation sampling with live-points, in: Proceedings of 2006 IEEE International Symposium on Performance Analysis of Systems and Software, ISPASS, 2006, pp. 2–12.

[20] N. Nikoleris, D. Eklov, E. Hagersten, Extending statistical cache models to support detailed pipeline simulators, in: Proceedings of 2014 IEEE International Symposium on Performance Analysis of Systems and Software, ISPASS, 2014, pp. 86–95.

[21] N. Nikoleris, L. Eeckhout, E. Hagersten, T.E. Carlson, Directed statistical warming through time traveling, in: Proceedings of the 52nd Annual IEEE/ACM International Symposium on Microarchitecture, MICRO, 2019, pp. 1037–1049.

[22] https://www.spec.org/cpu2006/.

[23] https://buildroot.org/.

[24] T.E. Carlson, W. Heirman, L. Eeckhout, Sampled simulation of multi-threaded applications, in: Proceedings of 2013 IEEE International Symposium on Performance Analysis of Systems and Software, ISPASS, 2013, pp. 2–12.

[25] T. Grass, T.E. Carlson, A. Rico, G. Ceballos, E. Ayguade, M. Casas, M. Moreto, Sampled simulation of task-based programs, IEEE Trans. Comput. 68 (2) (2018) 255–269.

BenchCouncil Transactions on Benchmarks, Standards and Evaluations 2 (2022) 100086

Contents lists available at ScienceDirect

BenchCouncil Transactions on Benchmarks, Standards and Evaluations

journal homepage: www.keaipublishing.com/en/journals/benchcouncil-transactions-on-benchmarks-standards-and-evaluations/

Case report

Edge AIBench 2.0: A scalable autonomous vehicle benchmark for IoT–Edge–Cloud systems

Tianshu Hao [a,b,*], Wanling Gao [a], Chuanxin Lan [a], Fei Tang [a,b], Zihan Jiang [a,b], Jianfeng Zhan [a,b]

[a] Research Center for Advanced Computer Systems, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
[b] University of Chinese Academy of Sciences, Beijing, China

ARTICLE INFO

Keywords:
IoT–Edge–Cloud
Benchmark
Autonomous vehicles
Scalable

ABSTRACT

Many emerging IoT–Edge–Cloud computing systems are not yet implemented or are too confidential to share the code or even tricky to replicate its execution environment, and hence their benchmarking is very challenging. This paper uses autonomous vehicles as a typical scenario to build the first benchmark for IoT–Edge–Cloud systems. We propose a set of distilling rules for replicating autonomous vehicle scenarios to extract critical tasks with intertwined interactions. The essential system-level and component-level characteristics are captured while the system complexity is reduced significantly so that users can quickly evaluate and pinpoint the system and component bottlenecks. Also, we implement a scalable architecture through which users can assess the systems with different sizes of workloads.

We conduct several experiments to measure the performance. After testing two thousand autonomous vehicle task requests, we identify the bottleneck modules in autonomous vehicle scenarios and analyze their hotspot functions. The experiment results show that the lane-keeping task is the slowest execution module, with a tail latency of 77.49 ms for the 99th percentile latency. We hope this scenario benchmark will be helpful for Autonomous Vehicles and even IoT–edge–Cloud research. Now the open-source code is available from the official website https://www.benchcouncil.org/scenariobench/edgeaibench.html.

1. Introduction

As a typical complex real-world application, IoT–Edge–Cloud systems consist of "a diversity of AI and non-AI modules with huge code sizes and long and complicated execution paths" [1]. Moreover, many emerging IoT–Edge–Cloud computing systems are yet implemented or are too confidential to reveal their technical details, not to mention sharing the source code. For example, a typical IoT–Edge–Cloud system – autonomous vehicles – runs 100 million lines of code in just one car [2]. Overall, they are too tricky or costly to replicate the code or even their execution environments; hence, their benchmarking is very challenging.

Even if we can replicate the application completely, directly using the application as the benchmark have several pitfalls. Real-world applications often have many instantiation biases. That is to say, real-world applications or systems are trapped in limited design and implementation points in a high-dimension space [3]. Previous work [4] has discussed the root cause of the instantiation bias. A workload is hierarchically implemented in a modern computer system: a problem definition, an algorithm, an intermediate representation, an ISA-specific representation, and a micro-architectural representation. From top to down, the design and implementation spaces increase explosively. However, for maintaining user experience or saving the software and hardware ecosystem investment, users adhere to existing products, tools, platforms, and services, which the previous work called technology inertia [3,5]. The technology inertia traps the real-world solution to a problem into a specific exploration path — a subspace or even a point at a high-dimension solution space. While profiling has been applied to various aspects of benchmarking complex real-world applications [1], the profiling technique helps little in overcoming this limitation.

Gao et al. [1] proposed a scenario benchmarking methodology to attack the above challenge. They proposed several rules to distill a real-world application scenario from a high-level requirement specification into a combination of essential AI and non-AI tasks as a scenario benchmark. Meanwhile, They identify primary modules in the critical paths of a real-world scenario from the system implementation level as they consume the most system resources and are the core focuses for system design and optimization. However, they fail to consider the complex IoT–Edge–Cloud scenarios. This paper extends the scenario benchmarking methodology for IoT–Edge–Cloud systems. For the first time, Hao et al. [6] propose an end-to-end view in benchmarking IoT–Edge–Cloud systems, considering all three layers: client-side devices,

* Corresponding author at: Research Center for Advanced Computer Systems, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China.
E-mail addresses: haotianshu@ict.ac.cn (T. Hao), gaowanling@ict.ac.cn (W. Gao), lanchuanxin@ict.ac.cn (C. Lan), tangfei@ict.ac.cn (F. Tang), jiangzihan@ict.ac.cn (Z. Jiang), zhanjianfeng@ict.ac.cn (J. Zhan).

https://doi.org/10.1016/j.tbench.2023.100086
Received 20 November 2022; Received in revised form 13 February 2023; Accepted 13 February 2023
Available online 16 February 2023

edge computing layer, and cloud servers. But their methodology has flaws. For example, they fail to consider the problem definition and instantiation bias. Moreover, they only implemented several component benchmarks in isolation without realistic interactions, which cannot constitute an end-to-end view. In addition, their workloads are not scalable.

The autonomous vehicles case is selected by most state-of-the-art benchmarks as a representative scenario and has the typical features of IoT–Edge–Clod systems [7–11]. Moreover, autonomous vehicles may be the most safety-critical scenario because it is crucial to human life. Therefore, building a unified, reasonable, and general benchmark set for autonomous vehicles is essential. There are several benchmarks for autonomous driving, such as KITTI [12], CAVBench [11], and Chauffeur [13]. However, they lack the scenario view to construct the benchmark, which will lead to a lack of the performance of the whole system. Therefore, in this paper, we choose autonomous vehicles as the case study to create a scalable scenario benchmark for IoT–Edge–Cloud systems, benefiting users to evaluate and improve their systems and applications.

The autonomous vehicles scenario is highly complex. Various AI vision workloads and critical decision tasks are presented in an autonomous car. These numerous tasks generate substantial input data, and these premises bring uncertainty to system function [14]. The Society of Automotive Engineers (SAE) classifies the quality of automation of a system into six levels of autonomous driving systems; the higher the level, the higher the system's performance, with L5 representing full automation [15]. While today's most advanced autonomous driving systems rarely reach L4 and L5, industrial companies are more focused on developing L2 and L3 level technologies, and there are still specific bottlenecks in the current level [16]. In summary, it is crucial to establish a unified, reasonable, and general benchmark set for autonomous driving, which will benefit the research and development of systems and applications in the field of autonomous vehicles.

In this paper, based on the state-of-the-art benchmarking methodology [1,3,4], we select autonomous vehicles as a research case to establish a scalable scenario-l benchmark for IoT–Edge–Cloud systems, which reduces the complexity of the system while maintaining the typical characteristics and critical execution path. This benchmark facilitates users in evaluating the system's performance and improving the algorithm. Finally, we conduct experiments using this IoT–Edge–Cloud scenario benchmark to analyze the critical task workloads in autonomous driving.

We sum up our main contributions as follows:

1. In order to ensure that the system's features are preserved as much as possible during the distilling process, we propose six distilling rules to simplify the scenario based on the characteristics of autonomous vehicles.
2. We propose the first scenario benchmark for the IoT–Edge–Cloud systems and provide the reference implementation. In addition, we also implement a scalable framework to support different sizes of workloads.
3. In the experiment section, we test two thousand autonomous vehicle tasks the end devices sent and measure the tail latency of each module. The results show the slowest execution module is the lane-keeping task and the convolution operations are the hotspot functions. Therefore, a scenario-based benchmark will help users find the bottleneck module of a system.

We organize the rest of this paper as follows. Section 2 summarizes the complexity of the autonomous vehicle scenario, the problem definition and instantiating in benchmark construction, and related work. Section 3 introduces the construction of scenario-level benchmarks. Section 4 introduces our scalable edge computing architecture. Section 5 performs evaluation. Section 6 concludes.

2. Problem definition and solution instantiation

Zhan [3] points out that a benchmark needs three processes: problem definition, solution instantiation, and measurement. We follow this guide to build our benchmark. Due to the different IoT, Edge, Cloud systems, workloads, and performance requirements, defining and instantiating the problems and their solutions is challenging.

2.1. Problem definition

Initially, this paper focuses on the problem of how to help users to get better performance from an IoT–Edge–Cloud system. Thus we extract a few critical workloads and provide a scalable benchmark to evaluate the systems to meet the performance requirements.

Secondly, we take autonomous vehicles, the most representative IoT–Edge–Cloud scenario, as the case study in this paper. Like most IoT–Edge–Cloud scenarios, autonomous vehicles have many complexity and entanglement among different components of the architecture and workloads. A comprehensive autonomous vehicle system may include many processing tasks. The driving automation is taken into six levels according to the international standard SAE J3016 with reference to the performance of the dynamic driving task (DDT) on a sustained basis [15]. Therefore, we take the DDT applications as the primary concern to instantiate the problem.

Thirdly, we extract the representative workloads and formalize them with a directed acyclic graph-based (DAG) model. Then we distill their critical path to build a scenario benchmark by the scenario benchmarking methodology [1]. We design and provide a workload reference implementation that reflects the characteristics of real scenarios.

To meet different users' requirements of the scales, we design and implement a scalable benchmark based on the scenario benchmark framework—scalable architecture helps the system allocate resources and workloads.

2.2. Complexity of autonomous vehicles scenario

Like most IoT–Edge–Cloud scenarios, an autonomous vehicle system has numerous application-level components. These components carry out a lot of communication across three layers of IoT–Edge–Cloud system architectures, making the system more complicated. The distributed three-layer architecture needs computing resources scaling and workload allocation of multiple layers. Accordingly, the scalability of the benchmark is also important to adapt to different sizes of workloads and meet the performance requirements of different users. However, unlike other scenarios, an autonomous vehicle system has its own characteristics. We summarize them below.

1. **The system sophistication .** The entire autonomous vehicle system involves a wide range of communications and data interaction. Meanwhile, it is also filled with numerous perception, planning, decision-making, and other autonomous driving tasks. The entire system processes massive volumes of data while running those intricate AI and non-AI algorithms in real time. Different design strategies provide difficulties for both hardware and software systems.
2. **Varied environmental factors.** During the driving process, the car will encounter various natural weather conditions (e.g., fog and snow) and complex terrain factors (e.g., mountains and hills), which will impact sensor data collection and the accuracy of AI tasks like objection recognition. Additionally, the existing autonomous driving system may not have an accurate judgment in extreme weather [17]. Hence, a reliable autonomous vehicle system must take into account a variety of weather conditions.

3. **Massive amount of input data.** Autonomous cars are equipped with many sensors, GPS positioning modules, and cameras for data collection, which will generate a large amount of heterogeneous input data [18] constantly. Moreover, multiple onboard cameras will keep collecting information about the surrounding environment. The system needs to consider how and where this data is processed, stored, and trained.
4. **The high demand for accuracy.** Automated driving tasks require absolutely correct decisions from the autonomous driving system. However, many tasks in the present AI models cannot achieve accuracy above 90% [19]. Additionally, there will be more uncertainties in the autonomous driving environment, such as sudden braking of the vehicle in front, pedestrians entering the road, and other unexpected situations. Consequently, safety can also be achieved during stable driving.
5. **Stable network performance.** As a typical IoT–Edge–Cloud system, an autonomous vehicle system requires real-time data interaction with cloud data centers and edge servers throughout the entire vehicle network. To support this, a high-bandwidth and high-performance network environment is required.
6. **Limited computing resources.** Real-time task processing has high requirements for computer resources due to the enormous amount of data. However, the processing capability of in-vehicle chips is constrained. Therefore, it needs to develop a lightweight model for these AI tasks to match the in-vehicle computing system is a significant issue. Numerous improved AI model pruning techniques [20,21] are now being presented to overcome the obstacle and meet the real-time requirement.
7. **High energy consumption.** Autonomous vehicles are equipped with numerous sensors and powerful processing chips, which have high energy consumption. According to studies, the overall power consumption of cars will rise by 2.8 to 4 percentage points to enable self-driving capabilities [22]. With the development of 5G technology, the energy demands of network communication will increase.

As summarized above, real autonomous driving systems are pretty complicated, making it difficult to fully and accurately model these characteristics in creating a representative benchmark.

2.3. Related work

In recent years, the field of autonomous vehicles with AI technologies has started to acquire traction. There is some relevant benchmarking research work for autonomous driving.

KITTI [12] is a vision benchmark suite for autonomous driving. It proposes stereo and optical vision data collected from the camera and the laser scanner. However, its purpose is to evaluate the vision algorithms' performance, not the whole autonomous driving system.

CAVBench [11] is the first benchmark suite for edge computing systems. It summarizes four scenarios and implements six AI workloads for autonomous vehicles. It takes an end-to-end view considering edge computing architecture. Nevertheless, it lacks a whole scenario-level view.

Chauffeur [13] is an open-source benchmark for autonomous driving. It implements end-to-end pipelines considering sensing, planning, and actuation processes. But it also did not consider the whole picture of the autonomous vehicle based on end-edge-cloud three-layer architecture.

In conclusion, the state-of-the-art autonomous vehicle benchmarks lack the scenario-level view to consider the whole scenario picture. They concentrate on specific algorithms, AI workloads, or hardware performance. However, an autonomous vehicle scenario is typical in IoT–Edge–Cloud systems, which must consider the components and the whole system's performance. Therefore, we need to distill the key module of the system and create a new scenario benchmark to simulate the real-world system's performance.

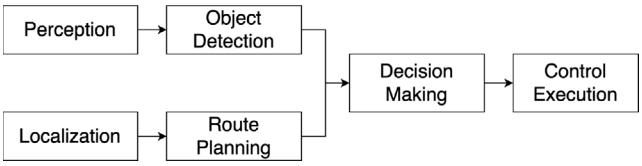


Fig. 1. Task flow chart of autonomous vehicles.

3. Creating the scenario benchmark

Based on the above challenges and motivations, we present the methodology and construction process for building an autonomous vehicle scenario-level benchmark for an intelligent edge computing system.

3.1. Specifying the autonomous vehicle scenario

An autonomous driving system mainly consists of a three-layer processing structure of perception, decision planning, and control execution module. The perception and control layers can be secured by configuring multi-layer redundant hardware systems. Thus, in the current research on autonomous driving systems, we focus mainly on the core algorithms for decision planning. The main emphasis in creating scenario benchmarks is likewise on decision planning-related artificial intelligence task modules.

According to the grading table of the international standard SAE J3016, it classifies driving automation into six levels with reference to the automation of dynamic driving tasks (DDT), DDT fallback, and object and event detection and response (OEDR) tasks on continuous driving systems. OEDR is a subtask of the DDT, which includes real-time object identification, classification, and other AI tasks. When a dynamic driving task fails, the system must perform the DDT fallback [15]. As a result, we instantiate our benchmark problem with the dynamic driving task as the primary concern. As dynamic driving is the fundamental task of autonomous driving, according to which we classify typical dynamic driving tasks and summarize the scenarios of autonomous vehicles in IoT–Edge–Cloud systems.

As shown in Fig. 1, the workflows of a complete set of dynamic tasks for autonomous driving include perception, location, path planning, object detection, and final decision-making. The perception module collects data through sensors and cameras, the localization module combines GPS module and map information to locate the vehicle's position, and the route planning module carries out a path planning task to determine an appropriate driving route according to the user's destination. The recognition task contains the recognition of vehicles, roads, pedestrians, obstacles, and traffic sign lights [23]. Finally, based on these parallel tasks, the vehicle-centric processor makes judgments regarding the current situation and decides to control the vehicle physically.

An autonomous vehicle currently has an intelligent edge chip with deep learning model processing capability, which can handle common lightweight AI autonomous driving tasks in real-time. However, it still needs to collaborate with cloud data centers and edge servers to execute tasks during the vehicle driving process better. In summary, autonomous vehicles use three layers of IoT–Edge–Cloud system resources to carry out diverse tasks.

As shown in Fig. 2, we used a set of directed acyclic graph (DAG) models to formalize the overall autonomous vehicle's tasks.

Large computing tasks or tasks with low real-time requirements are usually offloaded to the cloud data center for execution. At the same time, the cloud data centers also execute the task of offline training and ongoing retraining of the model. In the Internet of Vehicles, the cloud data center must communicate with all vehicles and make the whole vehicle network scheduling decisions.

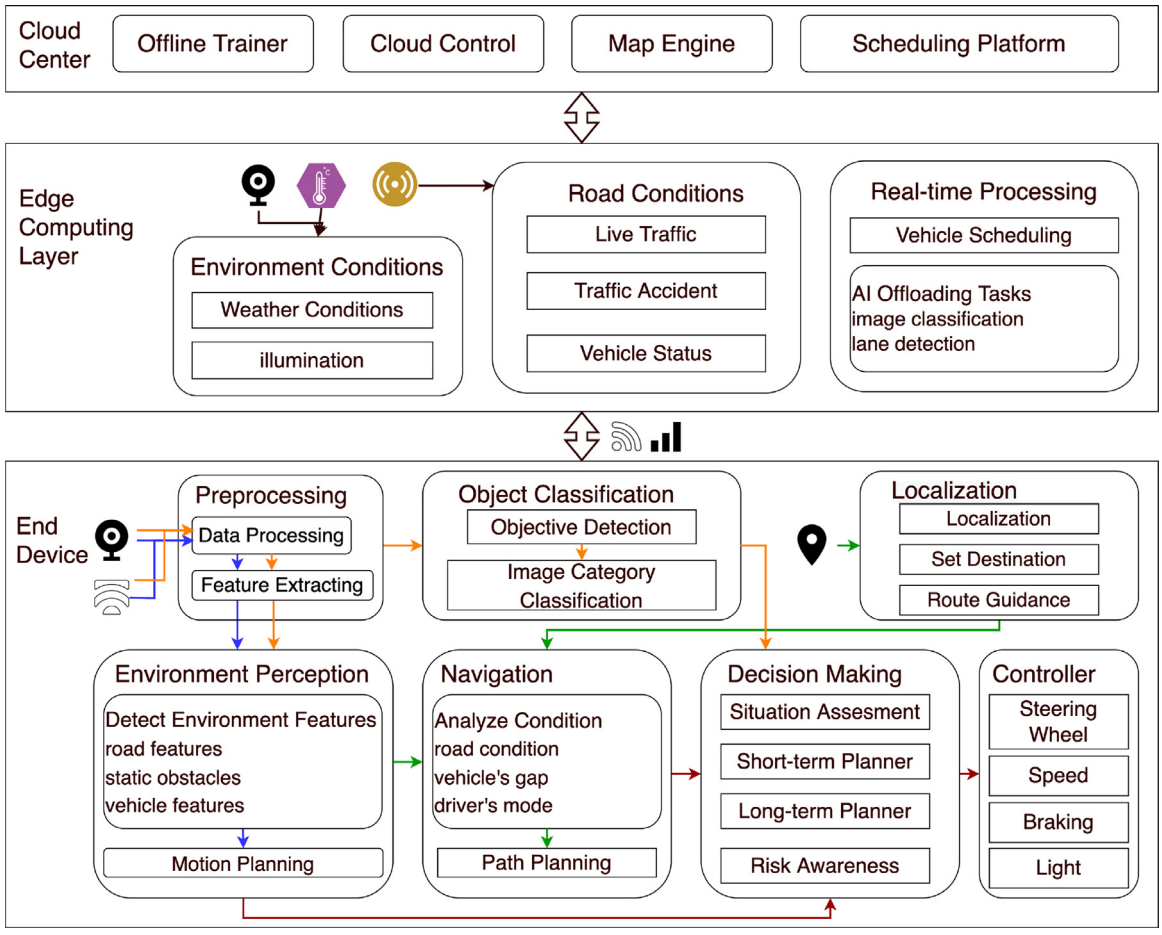


Fig. 2. The DAG model of autonomous vehicles in an IoT–Edge–Cloud system.

Autonomous vehicles connect to edge servers nearby when they move. These edge servers gather roadside environmental data, road information, and near-end vehicle data in real time, data that the vehicle's sensors often cannot collect because of blind spots and other issues. And the edge server will deliver it to nearby vehicles in a local area network. At the same time, with the guarantee of communication, autonomous driving vehicles will send tasks that cannot be processed in real-time by the onboard chip to the edge servers with sufficient computing power for processing. An excellent way to deal with issues like heterogeneous computing and energy consumption in autonomous vehicles is to offload jobs to the edge computing layer.

The smart chip on the vehicle side handles the primary autonomous driving workflow. The route planning and navigation tasks are executed by the vehicle in accordance with the user's instructions and GPS location data. In this procedure, the vehicle's sensors and cameras will gather data in real-time and pre-process them at the vehicle's end so that it can constantly recognize the environment, conduct perception tasks, and detect objects. The vehicle will simultaneously receive data from the edge server and cloud data center for integration. Finally, the vehicle's decision-making module will make decisions based on the information feedback from different modules and finally send the control commands.

3.2. Distilling rules for autonomous vehicles scenario

From Fig. 2, it is clear that formalizing the whole IoT–Edge–Cloud scenario is very complex. If the scenario benchmark is implemented accordingly, it will generate hundreds of millions of lines of code [24] and a vast amount of data, which is not conducive to users evaluating the system. Therefore, this section simplifies the autonomous vehicle scenario to extract several interdependent execution modules. Our work is inspired by the previous work [1] on the distilling rules for complex scenarios. And hence, the distilled modules can perform the critical tasks of an autonomous driving system while retaining the complexity and challenge of the system.

First, we propose a set of distilling rules for autonomous driving tasks based on real-world experience with autonomous driving, with reference to the industry's autonomous driving benchmark [11,12].

1. Retain only representative tasks among those that make use of similar models and serve similar purposes.
   In the process of autonomous driving, there are various types of object recognition and detection tasks, which include obstacle recognition, pedestrian recognition, traffic signal recognition, route recognition, etc. Most activities also share similar processing logic and critical path and are typically completed in two steps: detection and classification, with the exception of road route recognition in lane keeping. First, the object's location needs to be detected and localized in the video image, and then classification is performed to complete the recognition of the object. Therefore, we extract the critical traffic signal recognition from these tasks to ensure driving safety and the lane detection task to ensure vehicles obey traffic rules.
2. Prune the tasks executed on the cloud and edge servers and those in parallel with the user-end tasks.
   In IoT–Edge–Cloud systems, the user-end devices, edge servers, and servers in the cloud data center execute tasks in parallel and do not affect each other. Therefore, the training and scheduling tasks on the cloud do not affect the vehicle driving process. As a result, we prune this part of the tasks. At the same time,

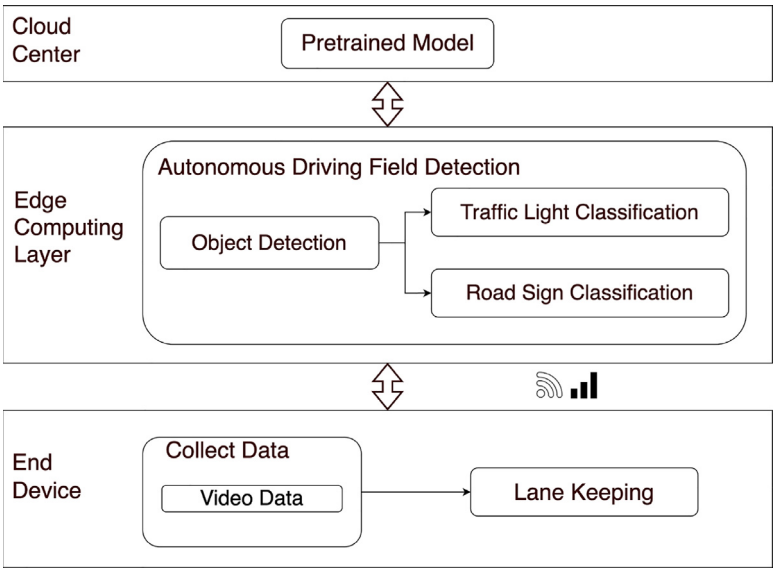


Fig. 3. The DAG model of the scenario of the autonomous vehicle in IoT–Edge–Cloud systems after simplifying.

the vehicle analysis and environment perception modules at the edge are also pruned.

3. Prune the modules whose running time is less than 1% of the total running time.
   After the analysis of the real system, the text data transmission latency and the specific processing time of the data collected by sensors, radar, and other devices occupy a very short period of time. The final task decision and control modules, which do not involve AI models, can also be completed in a very short time. Therefore, we will trim these modules.
4. Combine similar tasks that are executed concurrently if possible.
   In the traffic signal classification and road sign classification tasks, both have object detection for object localization. Thus, we merge the object detection process in these two modules. The results are sent to the subsequent tasks—traffic signal classification and road sign classification.
5. Remove the route planning module.
   Route planning is one of the most critical tasks in autonomous driving. But in creating this scenario benchmark, we remove it because the route planning task does not require real-time image data, and the panning result data transmission time is very short. This task is usually performed on a cloud server in existing real-world environments. The algorithms have been developed very maturely for the route planning task itself, and many advanced online navigation maps are available to users. Baidu's proposed Apollo autonomous driving level navigation [25] is now in use, reducing the speed of passing vehicles at intersections by 36.8%. As a result, this module can be trimmed.
6. Remove precedent and subsequent tasks of the pruning module.
   After simplifying the overall scenario according to the first five distilling rules, we will re-examine the DAG model and remove any prior or following tasks to the pruning module.

Based on the proposed six distilling rules, we prune Fig. 2 into a simplified DAG model 3. First, we merge similar modules with the same purpose. Next, we prune the parallel tasks executed on the IoT–Edge–Cloud systems at the same time. Then, we prune the modules that consume a short time, such as preprocessing and decision-making. Next, we combine similar tasks executed concurrently, such as the object classification module. At last, we removed the navigation module and related tasks.

With this simplified scenario of autonomous driving, we have scaled down the amount of code and dataset, reducing the complexity of the scenario while retaining the characteristics. Therefore, users can still evaluate systems and components in which they are interested.

4. The reference implementation of the autonomic vehicle scenario benchmark

4.1. Reference implementation

We investigated advanced algorithms and real-world datasets from academia and industry for the simplified autonomous driving scenario model proposed in the previous section. Then we implement the scenario-level benchmark for autonomous vehicles according to Fig. 3. This section briefly describes the deep learning algorithm models and datasets used in our reference implementation.

The **lane keeping** task used a CNN model [26] based on a self-attention distillation mechanism and selected CuLane [27] as a real-world dataset, which contains 3268 well-labeled training data and 358 validation data.

The **object detection** task uses YOLOv5 [28] as the deep learning network model and selected BDD100K [29] as the dataset, which contains 100,000 labeled HD datasets.

The **traffic Light classification** task uses a CNN model [30] as the deep learning network model. It uses the Nexar dataset [31] as the real-world dataset, which contains 18,659 labeled training datasets containing traffic signal images and 500,000 test data images.

The **road sign classification** task uses a CNN deep learning model [32] based on the LeNet framework [33] and the German Traffic Sign Recognition Benchmark (GTRB) [34] as the dataset, and the task classifies 43 classes of traffic signs.

4.2. A scalable IoT–Edge–Cloud benchmarking framework

In order to meet the real-world edge computing scenario, the benchmark architecture needs to consider resource allocation to deal with different sizes of AI workloads. For our simplified autonomic vehicle scenario, we propose a scalable architecture that can evaluate different sizes of systems and allocate resources (see Fig. 4).

This scalable architecture is based on Google Kubernetes [35], which allocates the offloading workloads to the edge server. On the edge server, we use TensorFlow Serving [36] to load the pre-trained models sent down from the cloud datacenter, waiting for the response to end-user tasks.

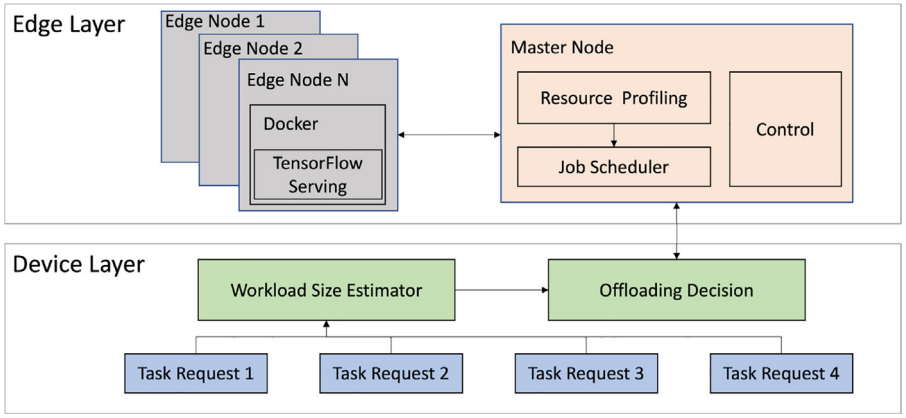


Fig. 4. A scalable IoT–Edge–Cloud benchmarking framework.

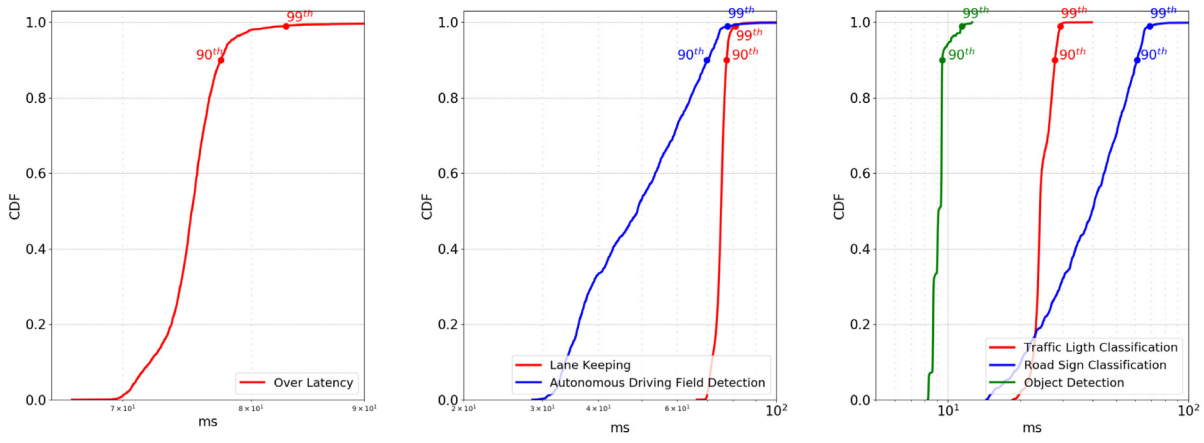


Fig. 5. Overall scenario and components latency breakdown of multi-tasks.

Table 1
Configurable parameters of the scalable framework.

Parameters	Description
The number of nodes	the number of edge computing layer nodes
AI module location	where the AI task placed: edge computing layer or end device
The number of tasks	the number of tasks that the device will send
Task size	the data input size of the task (MB)

In order to achieve system scalability, a master node is present at the edge layer to manage the computing resources and allocate workloads supplied by end devices. This architecture can scale numerous edge nodes to distribute tasks from end devices to various edge servers and computing resources. The parameters users can set are listed in Table 1.

5. Experiments and measurements

We conduct a scenario benchmark evaluation experiment based on a four-node server cluster, including one cloud server, two edge servers, and one client device. One experiment device is a CPU cloud server with two Xeon E5645 processors and 32 GB of RAM, and the other three nodes are each equipped with an Nvidia Titan XP GPU. Each node is connected to the other with a 1 GB Ethernet connection. We perform the offline training for the four AI tasks on the cloud servers and send the pre-trained model to the edge servers and end devices.

5.1. Tail latency of the whole scenario

Autonomous driving scenarios are very demanding in terms of latency, so we choose latency as a quality of service metric for this scenario benchmark. We break down the whole scenario latency to each task module to discover which modules are the primary contributors to latency in the whole scenario.

We tested 2000 autonomous vehicle task requests sent by the client device. Fig. 5 shows the latency of the whole scenario compared to the breakdown latency of each module. Since several vehicles send requests simultaneously in the real-world scenario, the tail latency metric is an important metric we need to concern about. We also pay attention to the latency data for the 90% and 99% percent of vehicle-side user queries.

Fig. 5(a) shows the end-to-end latency data for the entire scenario, with a tail latency of 76.45 ms for the 90th and 77.49 ms for the 99th percentile latency. In Fig. 5(b)(c), we have decomposed the tasks according to whether it belongs to the edge layer or the vehicle end. We can see that the overall latency of the lane-keeping task is slower than detection and classification tasks. And further decomposition of the object classification task shows that the slowest module is the road sign classification, with a tail latency of 58.92 ms for the 90th percentile latency and 67.70 ms for the 99th percentile latency. The fastest task

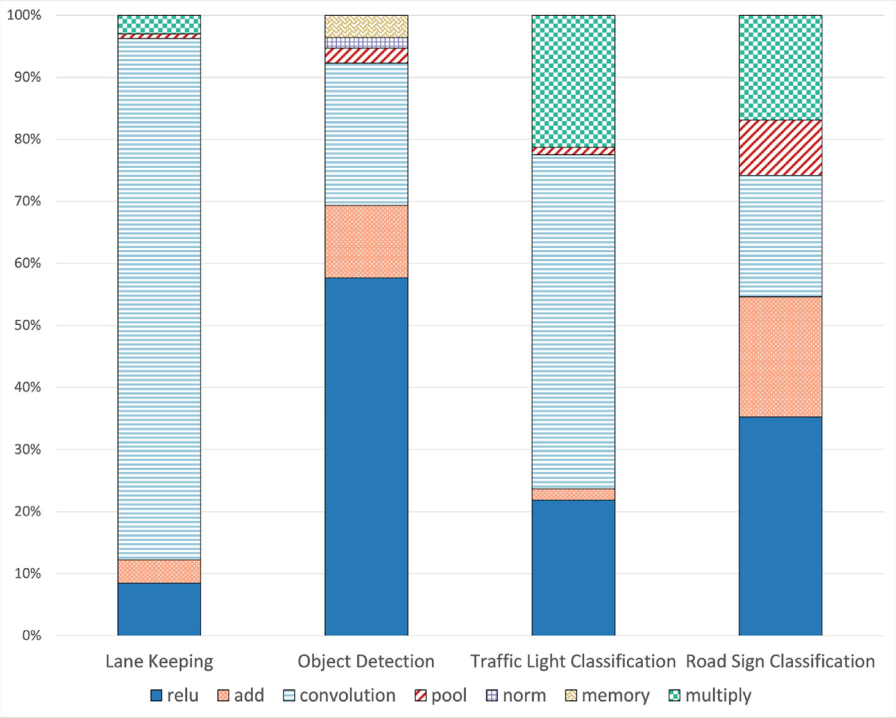


Fig. 6. Hotspots functions runtime breakdown.

is the object detection task, with a tail latency of 8.67 ms for the 90% task and 9.40 ms for the 99% task. Moreover, for the traffic signal classification, the 90th percentile latency is 23.55 ms, and the 99% percentile latency is 28.13 ms.

In our scenario benchmarking framework, the lane-keeping task is placed on the vehicle side. However, the large AI model of this task causes a slow processing speed. Therefore, it reduces the system's overall execution speed. According to the network environment's performance, users can try to place lane-keeping tasks on the edge side. An appropriate task position strategy will achieve better performance.

5.2. Hotspot function analysis

Because the majority of the tasks in autonomous driving scenarios employ deep learning models, which require the high performance of the onboard chips, as a result, we decomposed the execution time of GPUs using the profiling tool nvprof [37] offered by Nvidia. Then we analyze those hotspot functions.

We analyze each module's runtime using the nvprof tool to identify the hot functions that consume the most runtime. Then we divide these functions into seven categories based on their intrinsic computational logic: ReLU activation functions, add operations, convolution operations, pool operations, normalization, memory operations, and matrix multiplication operations.

As can be seen in Fig. 6, the convolution operations account for the most time in the lane-keeping task, which is why it is the slowest execution module. Additionally, the convolution operations take up a large percentage of all other tasks.

In object detection and road sign classification, the function with the most execution time is the ReLu activation function.

Analyzing the hotspot function is beneficial to further optimizing the CUDA library of the smart chip in autonomous vehicles. Also, for the special scenario of autonomous driving, software and hardware co-design is needed to optimize the execution speed of different modules and thus optimize the overall scenario performance.

6. Conclusion

This paper proposes the first IoT–Edge–Cloud benchmark: a scenario benchmark for autonomous vehicles. First, we analyze the challenges of creating an autonomous driving scenario benchmark. Then We reproduce the whole autonomous driving scenario picture under the IoT–Edge–Cloud system based on these user-concerned challenges and industrial-grade autonomous driving scenarios. Because of the specificity of the autonomous driving scenario, many complex factors must be considered. Therefore, the amount of code to reproduce the entire system based entirely on this scenario graph is enormous.

To resolve this issue, we propose to take a scenario benchmark view. We propose six distilling rules for simplifying the scenario of the autonomous vehicle. These rules ensure that the system's characteristics are retained while streamlining the whole system as much as possible and covering critical end-to-end IoT–Edge–Cloud paths. We obtained a simplified DAG diagram of the essential tasks from the autonomous vehicle scenario according to the distilling rules and implemented them with state-of-the-art techniques. To meet the system-level evaluation at different scales, we also implement a scalable Iot–Edge–Cloud benchmarking framework for the autonomic vehicle scenario.

Finally, we conduct several experimental evaluations of this scenario benchmark and measure the tail latency of each module. The experimental results show that lane-keeping is the most time-consuming task in the whole system. In addition, we make further analysis of the hotspot function. The result indicates that the convolution operation is the most time-consuming function. The experiment results reveal the optimization points for the software stack of autonomous vehicles.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This research was supported by the Strategic Priority Research Program of the Chinese Academy of Sciences, Grant No. XDA0320000 and XDA0320300.

References

[1] W. Gao, F. Tang, J. Zhan, X. Wen, L. Wang, Z. Cao, C. Lan, C. Luo, X. Liu, Z. Jiang, Aibench scenario: Scenario-distilling AI benchmarking, in: 2021 30th International Conference on Parallel Architectures and Compilation Techniques, PACT, IEEE, 2021, pp. 142–158.

[2] J. Somers, The coming software apocalypse, Atl. 26 (2017) 1.

[3] J. Zhan, A BenchCouncil view on benchmarking emerging and future computing, BenchCouncil Trans. Benchmarks, Stand. Eval. (2022) 100064.

[4] J. Zhan, Call for establishing benchmark science and engineering, BenchCouncil Trans. Benchmarks, Stand. Eval. 1 (1) (2021) 100012, http://dx.doi.org/10.1016/j.tbench.2021.100012, URL; https://www.sciencedirect.com/science/article/pii/S2772485921000120.

[5] J. Zhan, Three laws of technology rise or fall, BenchCouncil Trans. Benchmarks, Stand. Eval. 2 (1) (2022) 100034, http://dx.doi.org/10.1016/j.tbench.2022.100034, URL; https://www.sciencedirect.com/science/article/pii/S2772485922000217.

[6] T. Hao, Y. Huang, X. Wen, W. Gao, F. Zhang, C. Zheng, L. Wang, H. Ye, K. Hwang, Z. Ren, et al., Edge AIBench: towards comprehensive end-to-end edge computing benchmarking, in: Benchmarking, Measuring, and Optimizing: First BenchCouncil International Symposium, Bench 2018, Seattle, WA, USA, December 10-13, 2018, Revised Selected Papers 1, Springer, 2019, pp. 23–30.

[7] T. Hao, K. Hwang, J. Zhan, Y. Li, Y. Cao, Scenario-based AI benchmark evaluation of distributed cloud/edge computing systems, IEEE Trans. Comput. (2022).

[8] H. Blum, P.-E. Sarlin, J. Nieto, R. Siegwart, C. Cadena, Fishyscapes: A benchmark for safe semantic segmentation in autonomous driving, in: Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, 2019.

[9] X. Song, P. Wang, D. Zhou, R. Zhu, C. Guan, Y. Dai, H. Su, H. Li, R. Yang, Apollocar3d: A large 3d car instance understanding benchmark for autonomous driving, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 5452–5462.

[10] J. Xue, J. Fang, T. Li, B. Zhang, P. Zhang, Z. Ye, J. Dou, BLVD: Building a large-scale 5d semantics benchmark for autonomous driving, in: 2019 International Conference on Robotics and Automation, ICRA, IEEE, 2019, pp. 6685–6691.

[11] Y. Wang, S. Liu, X. Wu, W. Shi, CAVBench: A benchmark suite for connected and autonomous vehicles, in: 2018 IEEE/ACM Symposium on Edge Computing, SEC, IEEE, 2018, pp. 30–42.

[12] A. Geiger, P. Lenz, R. Urtasun, Are we ready for autonomous driving? the kitti vision benchmark suite, in: 2012 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2012, pp. 3354–3361.

[13] B. Maity, S. Yi, D. Seo, L. Cheng, S.-S. Lim, J.-C. Kim, B. Donyanavard, N. Dutt, Chauffeur: Benchmark suite for design and end-to-end analysis of self-driving vehicles on embedded systems, ACM Trans. Embed. Comput. Syst. (TECS) 20 (5s) (2021) 1–22.

[14] Y. Ma, Z. Wang, H. Yang, L. Yang, Artificial intelligence applications in the development of autonomous vehicles: a survey, IEEE/CAA J. Autom. Sin. 7 (2) (2020) 315–329.

[15] SAE, SAE J3016-taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles, 2021.

[16] H. Martin, K. Tschabuschnig, O. Bridal, D. Watzenig, Functional safety of automated driving systems: Does ISO 26262 meet the challenges? in: Automated Driving, Springer, 2017, pp. 387–416.

[17] N.A. Rawashdeh, J.P. Bos, N.J. Abu-Alrub, Drivable path detection using CNN sensor fusion for autonomous driving in the snow, in: Autonomous Systems: Sensors, Processing, and Security for Vehicles and Infrastructure 2021, Vol. 11748, SPIE, 2021, pp. 36–45.

[18] H. Daembkes, Automated driving safer and more efficient future driving foreword, in: Automated Driving: Safer and more Efficient Future Driving, Universität Ulm, 2017, pp. V–VI.

[19] L. Liu, S. Lu, R. Zhong, B. Wu, Y. Yao, Q. Zhang, W. Shi, Computing systems for autonomous driving: State of the art and challenges, IEEE Internet Things J. 8 (8) (2020) 6469–6486.

[20] H. Rebecq, T. Horstschäfer, G. Gallego, D. Scaramuzza, EVO: A geometric approach to event-based 6-DOF parallel tracking and mapping in real time, IEEE Robot. Autom. Lett. 2 (2) (2016) 593–600.

[21] Y. Cai, T. Luan, H. Gao, H. Wang, L. Chen, Y. Li, M.A. Sotelo, Z. Li, YOLOv4-5D: An effective and efficient object detector for autonomous driving, IEEE Trans. Instrum. Meas. 70 (2021) 1–13.

[22] J.H. Gawron, G.A. Keoleian, R.D. De Kleine, T.J. Wallington, H.C. Kim, Life cycle assessment of connected and automated vehicles: sensing and computing subsystem and vehicle level effects, Environ. Sci. Technol. 52 (5) (2018) 3249–3256.

[23] M. Simon, K. Amende, A. Kraus, J. Honer, T. Samann, H. Kaulbersch, S. Milz, H. Michael Gross, Complexer-yolo: Real-time 3d object detection and tracking on semantic point clouds, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 2019.

[24] P. Sagal, Bosch seeks edge with combined software, electronics unit. URL; https://europe.autonews.com/suppliers/bosch-seeks-edge-combined-software-electronics-unit/.

[25] Baidu, Apollo, 2020, URL; https://developer.apollo.auto/.

[26] Y. Hou, Z. Ma, C. Liu, C.C. Loy, Learning lightweight lane detection cnns by self attention distillation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 1013–1021.

[27] X. Pan, J. Shi, P. Luo, X. Wang, X. Tang, Spatial as deep: Spatial cnn for traffic scene understanding, 2017, arXiv preprint arXiv:1712.06080.

[28] H. Wang, Y. Xu, Y. He, Y. Cai, L. Chen, Y. Li, M.A. Sotelo, Z. Li, YOLOv5-Fog: A multiobjective visual detection algorithm for fog driving scenes based on improved YOLOv5, IEEE Trans. Instrum. Meas. 71 (2022) 1–12.

[29] F. Yu, W. Xian, Y. Chen, F. Liu, M. Liao, V. Madhavan, T. Darrell, Bdd100k: A Diverse Driving Video Database with Scalable Annotation Tooling, Vol. 2, No. 5, 2018, p. 6, arXiv preprint arXiv:1805.04687.

[30] Z. Ouyang, J. Niu, Y. Liu, M. Guizani, Deep CNN-based real-time traffic light detector for self-driving vehicles, IEEE Trans. Mob. Comput. 19 (2) (2019) 300–313.

[31] V. Madhavan, T. Darrell, The Bdd-Nexar Collective: a Large-Scale, Crowsourced, Dataset of Driving Scenes, Ph. D. Thesis, Master's Thesis, EECS Department, University of California, 2017.

[32] C. Zhang, X. Yue, R. Wang, N. Li, Y. Ding, Study on traffic sign recognition by optimized Lenet-5 algorithm, Int. J. Pattern Recognit. Artif. Intell. 34 (01) (2020) 2055003.

[33] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proc. IEEE 86 (11) (1998) 2278–2324.

[34] J. Stallkamp, M. Schlipsing, J. Salmen, C. Igel, The German traffic sign recognition benchmark: a multi-class classification competition, in: The 2011 International Joint Conference on Neural Networks, IEEE, 2011, pp. 1453–1460.

[35] B. Burns, J. Beda, K. Hightower, L. Evenson, Kubernetes: Up and Running, O'Reilly Media, Inc, 2022.

[36] C. Olston, N. Fiedel, K. Gorovoy, J. Harmsen, L. Lao, F. Li, V. Rajashekhar, S. Ramesh, J. Soyke, Tensorflow-serving: Flexible, high-performance ML serving, 2017, arXiv preprint arXiv:1712.06139.

[37] Nvidia, Nvidia profiling toolkit, 2022, URL; https://docs.nvidia.com/cuda/profiler-usersguide/index.html.

BenchCouncil Transactions on Benchmarks, Standards and Evaluations 2 (2022) 100089

Contents lists available at ScienceDirect

BenchCouncil Transactions on Benchmarks, Standards and Evaluations

journal homepage: www.keaipublishing.com/en/journals/benchcouncil-transactions-on-benchmarks-standards-and-evaluations/

Case report

An era of ChatGPT as a significant futuristic support tool: A study on features, abilities, and challenges

Abid Haleem [a], Mohd Javaid [a,*], Ravi Pratap Singh [b]

[a] Department of Mechanical Engineering, Jamia Millia Islamia, New Delhi, India
[b] Department of Mechanical Engineering, National Institute of Technology, Kurukshetra, Haryana, India

ARTICLE INFO

Keywords:
Artificial Intelligence (AI)
ChatGPT
Role
Features
Capabilities
Challenges

ABSTRACT

Open Artificial Intelligence (AI) published an AI chatbot tool called ChatGPT at the end of November 2022. Generative Pre-trained Transformer (GPT) architecture is the foundation of ChatGPT. On the internet, ChatGPT has been rapidly growing. This chatbot enables users to discuss with the AI by inputting prompts, and it is based on OpenAI's language model. Although ChatGPT is fantastic and produces exciting results for writing tales, poetry, songs, essays, and other things, it has certain restrictions. Users may ask the bot questions, and it will reply with pertinent, convincing subjects and replies. ChatGPT has now risen to the top of several academic agendas. Administrators create task teams and hold institution-wide meetings to react to the tools, with most of the advice being to adopt this technology. This paper briefs about the ChatGPT and its need. Further, various Progressive Work Flow Processes of the ChatGPT Tool are stated diagrammatically. Specific features and capabilities of the ChatGPT Support System are studied in this paper. Finally, we identified and discussed the significant roles of ChatGPT in the current scenario. The neural language models that form the foundation of character AI have been developed from the bottom up with talks in mind. This technology implies that the programme uses deep learning methods to analyse and produce text. The model "understands" the subtleties of human-produced natural language using vast amounts of data from the internet.

1. Introduction

Open Artificial Intelligence (AI)'s ChatGPT, introduced as a prototype in November 2022, has attracted the interest of engineers, social media users, business owners, authors, and students. Machine learning (ML) undoubtedly has the potential for good, despite many people's concerns towards ChatGPT. ML has influenced various sectors since it was widely adopted, enabling tasks like high-resolution weather predictions and medical imaging analysis [1–3]. ChatGPT has the potential to alter the way various professions are carried out. This chatbot can converse like a person since it was developed using OpenAI. Customers may begin using ChatGPT by creating a free OpenAI account. This technology may leverage user-generated data to enhance its training algorithms [4,5].

The paradigm changes in information access brought about by ChatGPT may benefit tag-holding industries, including education, research, journalism, mass communication, Information Technology (IT), retail, and many others. Various convincing writing may be produced quickly using generative AI technologies, which can then adapt the writing in response to feedback to make it more suited for the task. This has ramifications for a broad range of sectors, including marketing copy-required businesses and IT and software companies that may profit from the quick, generally accurate code produced by AI models. Additionally, organisations may employ generative AI to produce better technical items, such as upscaled copies of medical photos. Additionally, firms may seek new business prospects and the ability to provide more value with time and resources [6–8].

The organisation behind ChatGPT development has been active in this field for years. OpenAI focuses on initiatives to enhance AI's capabilities and investigate its social effects. While there are many ways to structure an essay, the rigour of mathematics presents its own unique set of problems. Given that there is often just one correct solution to various issues and ChatGPT can demonstrate its operation, a student might efficiently utilise it without the teacher ever knowing [9,10]. The effects of AI might be perceived on an aesthetic level even if it is utilised responsibly, such as to verify grammar or sentence structure. Students can be discouraged or afraid to take chances with their work if the bot asserts that one way to accomplish things is the proper way. Similarly, the student cannot experiment with structure or develop their voice if ChatGPT is used to develop the framework for an assigned essay or written piece [11,12].

* Corresponding author.
E-mail addresses: ahaleem@jmi.ac.in (A. Haleem), mjavaid@jmi.ac.in (M. Javaid), singhrp@nitkkr.ac.in (R.P. Singh).
URL: https://scholar.google.co.in/citations?user=rfyiwvsAAAAJ&hl=en (M. Javaid).

https://doi.org/10.1016/j.tbench.2023.100089
Received 25 February 2023; Received in revised form 2 March 2023; Accepted 3 March 2023
Available online 5 March 2023

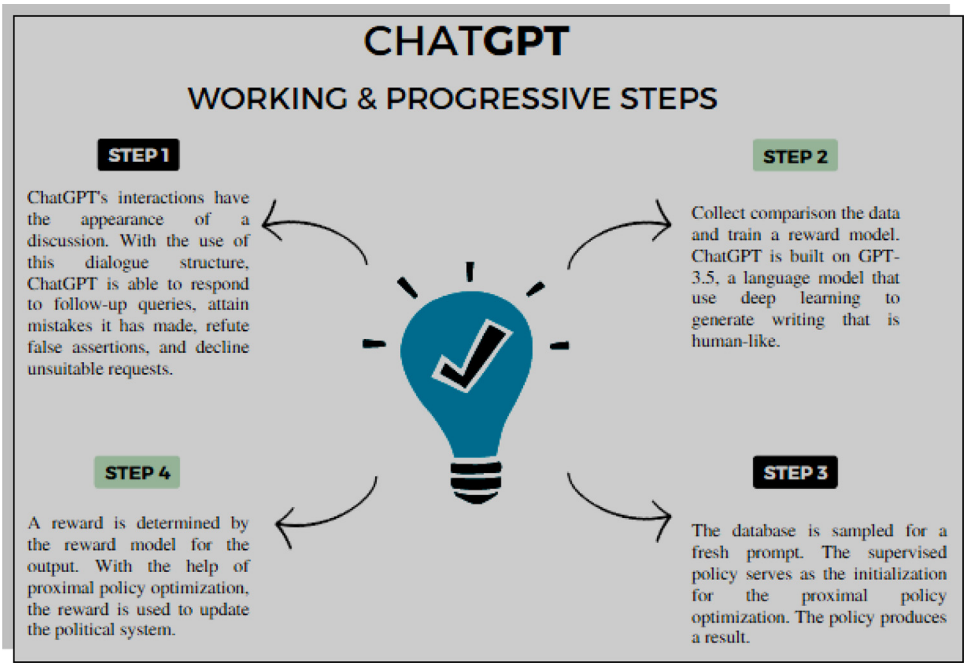


Fig. 1. Progressive steps involved in ChatGPT framework.

Deep learning is the most famous example of AI. In this technology, algorithms are trained on big datasets to generate predictions based on the data. It may include language translation, voice recognition, and picture recognition. AI that understands and produces human language is known as natural language processing. Translation, text summarisation, and sentiment analysis are some examples of this. The varied and creative AI models that ChatGPT utilises are based on unsupervised and semi-supervised ML methods. Images, extended text forms, emails, social media information, voice recordings, computer code, and structured data are just a few examples. They may also provide fresh material, translations, questions and answers, sentiment analysis, summaries, and movies. ChatGPT has the potential to advance a variety of spheres of our life, including healthcare, transportation, and education [13–15].

ChatGPT has quickly taken off on the global stage. Many people reading this need to know what this programme can achieve. This programme can produce unique articles about anything by employing an incredible capacity to fast search stuff online and powerful grammar and writing abilities. ChatGPT is a bot that has been taught to provide replies to user inputs that resemble a person's. It has developed a surprisingly broad range of talents using ML. On-demand, it can create elementary computer code, crude financial analysis, humorous poems and songs, perfect impersonations, reflective essays on just about any subject, summaries of technical papers or scientific ideas in natural language, chat-based customer service, accurate predictions, tailored guidance, and answers [16–18]. The primary research questions of this article are as follows:

RO1: - to brief about the ChatGPT and its need;
RO2: - to discuss the progressive workflow process of the ChatGPT tool;
RO3: - to study specific features and capabilities of the ChatGPT support system;
RO4: - to identify and discuss significant roles of ChatGPT in the current scenario;

2. ChatGPT

ChatGPT, a generative pre-trained transformer, is now attracting so much interest. The word "Generative" or "G" in the acronym GPT speaks for the tool's capacity to produce text. Pre-training, or "P", is the deployment of a model from one ML job to train another model, much as how individuals utilise prior knowledge to learn new things. ChatGPT offers a substantial amount of text to pre-train on. The neural network *T* is for "Transformer", which examines the overall connection between every data series component [19–21]. It is a free chatbot that can respond to practically any question. It was created by OpenAI and made available to the public for testing. It is already regarded as the finest AI chatbot ever. The chatbot has been known to produce computer code, college-level essays, poetry, and even half-decent jokes [22,23].

The first ChatGPT model was trained through supervised fine-tuning, in which human AI trainers conversed with both the user and an AI helper. The trainers have access to sample written recommendations to aid with answer composition. ChatGPT, a language model created expressly to comprehend and react to natural language, is one of their most recent innovations. This indicates that it can have natural and intuitive conversations with people. The best part of ChatGPT is that it is freely usable using OpenAI, which enables programmers to incorporate the model into their applications [24–26].

3. Progressive work flow process of ChatGPT tool

To process the ChatGPT working structure, a streamlined flow of information and knowledge is a must. Fig. 1 exemplifies the different working and progressive steps of the ChatGPT system for supporting the routine needs of the social structure. Various four steps are highlighted and discussed with the help of Figure. It started with the interactions and discussion, followed by the data reception and comparison creation. Further, the database gets sampled, and the process gets concluded by determining the reward model and updating the same in the cloud data set [27–29].

Business executives, students, and educators have a lot of potential opportunities to use this technology. Teachers who want to exhibit, explain, and have students apply concepts. A teacher could ask the student to describe their rationale and thinking process for the essay and then compare their explanation to the essay itself [30,31]. If there are significant differences between the essay and the student's explanation, it may indicate that the student employed a text-generation programme like ChatGPT. A teacher could also search for apparent

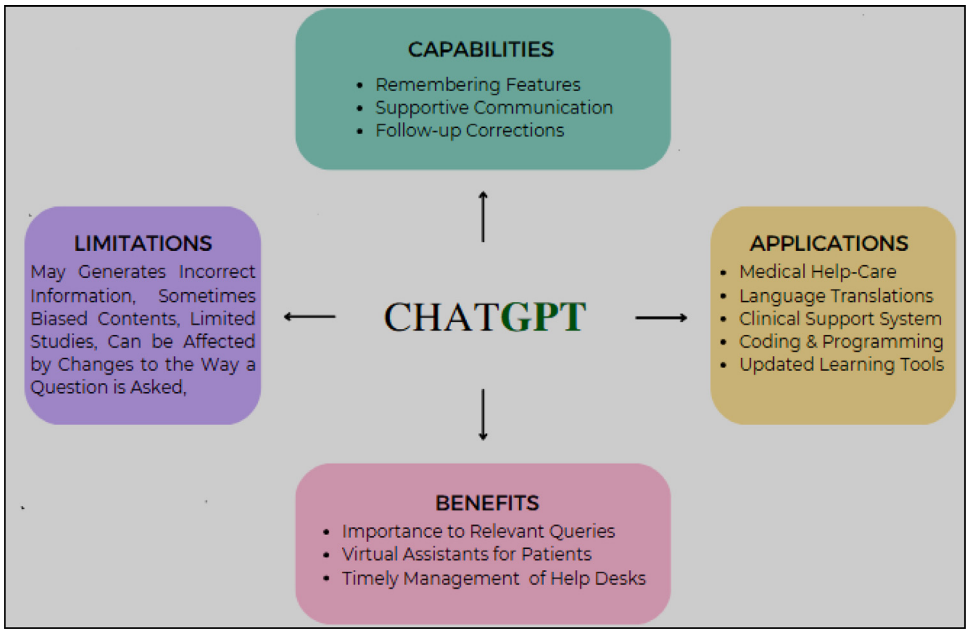


Fig. 2. Capabilities and features of ChatGPT.

indications of text production in the essay. It is common practice to employ ChatGPT for various jobs, from writing high school papers to creating legal documents and even composing legislation. These programmes can virtually instantaneously create more complex written material [32–34].

A teacher may determine if a student produced an essay independently or utilised a text-generation programme like ChatGPT in several ways. Utilising tools that can detect plagiarism is one way to see whether the essay is similar to any other. Text-generating programmes often provide output identical to existing content, and this may be a reliable technique to tell if a student used ChatGPT or another text-generation tool. ChatGTP engages in dialogue with the user, replies to follow-up inquiries, acknowledges and corrects errors, rejects inappropriate requests, and even questions false premises. ChatGPT is designed to respond promptly and thoroughly to instructions [35–38].

4. Specific features and capabilities of the ChatGPT support system

Fig. 2 explores the various associated capabilities, benefits, applications and limitations of ChatGPT support. It includes the features like remembering aspects, supportive communication, follow-up corrections, etc. Apart from these different features and capabilities, various limitations have been observed, such as; the sometimes generation of incorrect information, may rise with biased content, etc. In addition, several other benefits and applications of chatGPT are further represented and elaborated in Fig. 2 [39–41].

ChatGPT's ability to perceive the context and provide meaningful information makes it a helpful tool for collecting, evaluating, and understanding market trends. This technology can enhance current processes, gather qualitative data via informal surveys, analyse data and extract characteristics from vast volumes of unstructured data, give valuable market intelligence, and save researchers time and effort. Natural language processing is entering a new stage as early ChatGPT users show the technology's capacity to continue a discussion through several questions and create software code. Increasingly intricate interactions between people and machines are made possible by AI [42–44].

ChatGPT differs from previous AI models, as it can write software in many languages, debug code, break down a complicated subject into manageable chunks, prepare for interviews, and draft essays. ChatGPT simplifies such processes and even provides the result, much as how one might research similar subjects online. ChatGPT can produce texts that sound like human speech in an informal setting and perform basic tasks. ChatGPT aims to create a cooperative AI system that can produce language that is helpful, engaging, and contextually relevant [45–48].

5. Significant roles of ChatGPT in the current scenario

ChatGPT is to demonstrate and test the capabilities of a powerful AI system. ChatGPT is a generative AI programme that uses natural language processing to generate text, artwork, music, and video. A large language model powers ChatGPT, however, it needs data in order to function and develop over time. As a person, a model learns the training it receives. The algorithm becomes more adept at seeing patterns to predict future events and produce credible text [49–52]. It used a sequence model built for text production tasks, including question-and-answer, text summarisation, and machine translation. ChatGPT may provide suggestions for goods or information catered to specific needs and interests by studying client data. Businesses can develop distinctive experiences for new audiences, boost engagement, and build trust with the aid of ChatGPT. For companies looking to expand their customer base, access new markets, run efficient marketing campaigns, and forge closer bonds with both existing and prospective clients, ChatGPT may be a valuable tool [53–56]. The significant roles of ChatGPT in the current scenario are discussed in Table 1.

ChatGPT can determine what makes a company successful by examining its marketing tactics, clientele, product attributes, and other elements. It gives recommendations for how our company may adopt or enhance those characteristics. By examining market trends, the client wants, and other variables pertinent to our company, ChatGPT may suggest how to take advantage of these chances and expand our company by examining the product offerings, marketing initiatives, and customer engagement tactics of the rivals. Based on the target demographic, marketing objectives, and budget, ChatGPT may advise on the best channels for a specific campaign [57–59]. Social networking sites, email marketing, search engine marketing, and other digital marketing channels are examples of channels. Performance analytics assist organisations in tracking and evaluating the effectiveness of their digital marketing activities by revealing what works and what does not. To improve outcomes, the campaign plan and tactics may be modified in real-time using this information [60,61].

Table 1
Major roles of ChatGPT in the Current Scenario.

S No	Roles	Description
1.	Gaining widespread interest	• ChatGPT, the most cutting-edge AI language model, gained widespread interest. • It can create suggestions on almost everything, such as composing essays, articles, poetry, translated material, and more. • The AI classifier, a language model trained on a dataset of pairs of texts on the same subject produced by humans and by AI, tries to identify texts created by AI. • It has transformed how people engage with AI using its advanced natural language processing capabilities • ChatGPT has been trained on a vast quantity of text data and is very accurate at understanding and producing human-like replies to various themes. • ChatGPT is a formidable tool that can dramatically increase human productivity and creativity, whether used for answering queries, inspiring creative writing, or helping with daily work.
2.	Variety of language inputs	• Users may leverage ChatGPT's capacity to understand and react to diverse language inputs and obtain simple, uncomplicated answers to inquiries instead of using a search engine like Google • Getting a concise overview of important information is easier since the AI chatbot can explain complicated subjects in various speaking styles. • The training data for ChatGPT comes from the WebText dataset, an extensive collection of online text • This dataset contains a wide variety of text kinds and text styles, including articles, forums, and social media postings • By training on such a comprehensive dataset, ChatGPT can generate text equivalent to what people write • OpenAI's trained ChatGPT model can also analyse code and explain its purpose.
3.	Picking up of latest information	• The capability of ChatGPT to swiftly pick up on and adjust to new information is one of its main features • This indicates that it can handle new subjects and tasks without substantial retraining. • Furthermore, ChatGPT is very scalable, making it ideal for large-scale applications • ChatGPT may be used in various fields, including customer service, education, and entertainment • Natural Language Processing is one of its principal uses • The model is perfect for jobs like language translation, text summarisation, and question–answering since it can produce text depending on inputs. • Moreover, it has been used to develop chatbots and other conversational AI systems that may be applied to customer care and assistance applications • Generative AI may replace several occupations that can produce unique text, audio, and visual material in response to human input • The most popular notion is to employ 'assistants' like tools to make certain occupations more accessible to everyone.
4.	Learning and improving	• The capability of ChatGPT to gain knowledge from its interactions with users is one of its key advantage • It may modify and enhance its reactions when interacting with humans, gradually becoming more precise • The variety of use cases that ChatGPT can support through its adaptability makes it a potent instrument for further development and optimisation of conversational AI systems in the future • Experts predict that the success of ChatGPT will offer OpenAI a competitive edge over other AI firms • Although increasing use strains OpenAI's processing resources, it has also given vital input that has been used to refine the chatbot's replies. • This ChatGPT chatbot is trained on a vast quantity of text data from the internet by using a language model • ChatGPT has been trained using various textual materials, such as books, news stories, webpages, and more, giving it a comprehensive comprehension of various subjects • This model can comprehend the context and provide replies that are suitable for it.
5.	Helpful for a variety of tasks	• It may be used for various things, including creating code, recommending meals, and enhancing the quality of life for older people and those with impairments • The ability to utilise ChatGPT to complete assignments is available since every paper the bot creates is unique. • ChatGPT can react to a broad range of cues, almost nothing beyond its capabilities • The goal of the conversation GPT is to understand a simple statement • It provides us with guidance and assistance on right and wrong in the age of smartphones and computers. • It acts as a humanoid when we need to inquire about a different module since it will research to get the answers • The foundation model will significantly alter how software is developed and used across the technology sector, driven by platforms like the Role of ChatGPT.
6.	Respond to inquiries	• Several businesses are eager to integrate the ChatGPT AI-powered tool into their workflow to provide quick and knowledgeable solutions to frequent client inquiries and improve customer experience • The AI chatbot assists companies in quickly understanding and addressing consumer pain points by scanning the Internet for specific user inquiries and offering a brief overview of pertinent information • ChatGPT is skilled at answering questions, making recommendations, and making predictions • Software developers may use it to find and correct mistakes in their code • The fact that ChatGPT recalls the previous exchange may spur innovation and a surge in the popularity of personalised stress and therapy bots • For content moderation on decentralised social media sites, utilise ChatGPT • Examining the text and photos that users publish may, for instance, weed out spam and improper information.
7.	Business applications	• Businesses may develop more decisive marketing campaigns, engage with their target audience, and accomplish their marketing objectives when marketers use ChatGPT's capabilities. • There are several applications for ChatGPT and generative AI in business • As with almost any technological advancement, exercising caution is essential to ensure that private, sensitive, and secret business and personal information remains where it belongs. • Policymakers should be aware that the risks associated with AI systems created or deployed by various companies may be more significant, given the potentially high stakes for those impacted by choices • Processes search requests, gathers information from many sources, summarises papers, creates travel plans, responds to enquiries, and chats with people using OpenAI technology.

(*continued on next page*)

Table 1 (*continued*).

8.	Useful for digital marketers	• ChatGPT might be helpful for digital marketers that wish to enhance their campaigns and engage with their target consumers. • It helps create material for social media updates, blog entries, and other forms of content • The chatbot may recommend headlines, opening words, and even whole paragraphs for inclusion in marketing materials based on a subject or keyword • Digital marketers may better understand their target audience by doing audience research • ChatGPT can find the common traits, habits, and preferences of specific consumer groups by analysing massive amounts of data. • ChatGPT may be taught to respond to frequently asked queries, provide customer service, and suggest product • ChatGPT may be used to group customer reviews and unstructured data into categories based on product features, customer service, and marketing campaigns, improving analysis and understanding of consumer needs and preferences.
9.	Translate concepts	• ChatGPT has the ability to create computer code to create programmes and software • It has the ability to translate concepts from English into the programming language and verify human programmers' language for flaws. • The popularity of this technology is a new generation of generative models, mainly due to how approachable it is to the general public rather than its unique capabilities. • OpenAI's ChatGPT, a sizable language model, can create writing that resembles a person's • It can carry out various activities related to natural language processing, such as conversation systems, language summarisation, and translation • It is trained by using a large dataset of text from the internet.
10.	Better interpretability	• When more users supply information, the chatbot's interpretation skills will improve using reinforcement learning processes through human feedback • As a result, ChatGPT's response quality will advance over time to more effectively suit user demands • After that, the user experience will be enhanced as a result. To gather input and insights from consumers, • GPT may be utilised to design conversational and intelligent surveys. • It makes obtaining more accurate and exciting data possible than conventional surveys. • By examining vast amounts of customer feedback, social media postings, and other unstructured data sources, ChatGPT may be used to identify patterns in consumer views and behaviour.
11.	Genuine conversations	• ChatGPT is designed to look and sound like genuine conversations, and its responses seem very human. • When questioned, the bot can elaborate on ideas, recall what was stated previously in the dialogue, and even apologise when it makes a mistake • The foundation of ChatGPT technology employs supervised and unsupervised AI learning methods to train some of the most prominent language models in the world. • ChatGPT remembers all prior interactions, in contrast to most other chatbots • We may enter inquiries into ChatGPT using natural language, and the chatbot will respond with conversational responses gleaned from vast amounts of data from the internet and other public sources.
12.	Providing a dynamic and interesting way	• It is also a powerful tool to help kids write more dynamically and interestingly • Writing assignments may be better customised to meet the requirements and interests of each student with the help of AI-enabled technologies. • AI-enabled essay writing tools, for instance, may provide prompts in real-time and direct students through the writing process. • Even writing assignments that are customised to the student's interests and ability level may be generated with the aid of AI. • For instance, AI may provide writing prompts based on students' prior writing samples or themes they have previously liked writing about.
13.	Education	• ChatGPT can define words and sentences is impressive, which is helpful for education purposes. • When the chatbot's skills advance and are honed over the next few years, it may alter how students connect with the outside world. • ChatGPT use cases are in the education sector, where instructors may teach just the basics of a subject while providing students a forum to ask questions and clear up any confusion. • ChatGPT is a superior search engine to Google since it relies on the individual requirements and tastes of the user. • It could be the best option for people who prefer a conversational search experience over getting website links as search results.
14.	Developing stronger writing abilities	• It also helps with evaluation and grading, moving the emphasis from fixing mistakes to developing more vital writing abilities. • AI-enabled writing evaluation systems may automatically analyse and mark essays more precisely and effectively than human grading by using AI technologies like machine learning. • Students may further develop their writing talents by using AI systems to provide them with thorough feedback on their essays. • Developing distinctive, attractive, and appealing ad copy for various marketing initiatives may be difficult. • ChatGPT uses AI to produce practical text, which facilitates the work of a digital marketer. • This provides content concepts and structure to increase marketers' efficiency significantly.
15.	Create new things	• A generative AI system is made to create new things based on prior knowledge. • This technology is often created via a method known as machine learning, which entails instructing an artificial intelligence to carry out tasks by exposing it to a tonne of data, which it "trains" on and ultimately learns to duplicate. • For instance, ChatGPT has trained on a sizable amount of material from the internet and dialogue scripts to mimic real discussions. • ChatGPT will influence knowledge workers and the nature of employment in the future as AI becomes more pervasive. • ChatGPT may be used to evaluate customer reviews and determine the general sentiment of a brand, product, or service, providing essential insights into market research and relationship development.
16.	Cover a wide range of topics	• The artificial intelligence research organisation OpenAI's text-generating ChatGPT software can write about various topics in various prose and poetry styles. • It is capable of opining on itself. Similar to other chatbots, ChatGPT runs well. Users enter a question or "prompt" on the OpenAI website, such as "Suggest some prompts to try out a chatbot," • Shortly after, an AI-generated answer is returned. The software generates its responses using text prediction. • Its AI was trained on a large body of online human writing, enabling it to anticipate which word should come after the one before to give the impression of a rational being.

(*continued on next page*)

Table 1 (*continued*).

17.	Conduct routine duty at the office level	• The creation of virtual assistants for organisations that can conduct routine duties like making appointments, sending emails, and maintaining social media accounts may be done using ChatGPT. • This might be a wonderful method to automate repetitive operations, optimise workflow, and free up time for busy professionals so they can concentrate on more crucial duties like innovation and research. • OpenAI is at the forefront of generative AI, or technology trained on enormous volumes of text and photos that can generate content from simple language input. • AI has great promise for the creation of cutting-edge cybersecurity tools. Expanding AI and machine learning is essential to spotting possible dangers promptly. ChatGPT may be essential in identifying it, reacting, and enhancing internal communication during a cyberattack.

6. Discussion

AI has a straightforward concept and is now available for free access using OpenAI. A chatbot that can help us with a variety of activities is what we "speak" to. Due to its release, many companies are enthusiastic about utilising ChatGPT to simplify their procedures. Using ML algorithms that evaluate a vast amount of data and understand the patterns and structures of the language, ChatGPT is taught to produce text that resembles that of humans using a generative pre-trained transformer language model. Playing with OpenAI's ChatGPT has recently become quite popular. This artificially intelligent online correspondent will try to react to our inquiries with a paragraph's worth of information and create songs or tales in response to the instructions we provide. Because ChatGPT uses a conversation structure, it is possible to challenge false assumptions, admit mistakes, and reject unsuitable requests. This model can help with various Natural Language Processing problems, and because of its excellent scalability, it is perfect for usage in large-scale applications.

Although this technology has gained popularity over the years, mostly it still needs to be more basic and can only provide basic answers to inquiries on help desk sites or attempt to resolve the problems of dissatisfied consumers. The field of NLP is now gradually moving into a new chapter using ChatGPT's capability to continue a discussion through several questions and create software code. OpenAI intends to provide the tool as an application programming interface, enabling other parties to include it in their websites or applications without familiarity with the underlying technology. Therefore, businesses might soon employ ChatGPT to develop marketing tools, customer service bots, or virtual assistants. They might automate boring operations like document reviews. They may utilise it to produce fresh concepts and streamline decision-making.

Because ChatGPT can quickly compose material based on a prompt, it may be used to create content. ChatGPT may assist users in polishing their work and achieving their literary objectives. It can process, write, and assist in the debugging of code. Unstructured data is redundant because it is hard to handle, organise, and sort. ChatGPT saves the day since it can manipulate data to transform unstructured data into a structured manner. While most public interest in ChatGPT is focused on its text-generation capabilities, its capacity to comprehend that content may have the most significant commercial and social effect. AI uses statistical probability to create a model of the words and sentences that typically follow whatever text came before. It resembles predictive text on a smartphone, but it has been dramatically scaled up to create total replies rather than just single words.

7. Challenges in ChatGPT

According to OpenAI, ChatGPT may sometimes react to damaging instructions or display discriminatory behaviour and occasionally compose plausible-sounding but incorrect or nonsensical responses. It may also respond slowly, another issue that its creators blame. There is cause for excitement about such technologies, mainly if they help reduce obstacles to a better quality of life, such as the racial gaps in reading competence. On the other hand, there are a few techniques to reduce these dangers. Choosing the initial data used to train these algorithms is essential to prevent adding harmful material. Next, firms can consider utilising more minor, niche models rather than a generic generative AI model. Organisations with more significant resources might alter a generic model based on their own data to match their goals and reduce bias. Organisations should also avoid employing generative AI models for essential choices, such as those requiring significant resources or human well-being, and ensure that a real person reviews the output of a generative AI model before it is published or utilised.

AI can only partially address some of humanity's problems, where sustainability is essential. How well the data and techniques it is trained on are one of the significant AI constraints. As a result, AI systems are limited to making predictions based on the data they have been provided. Another drawback is that AI needs creativity, empathy, and other human-specific abilities. AI systems cannot conceive creatively or comprehend nuanced human emotions since they are only meant to carry out specified jobs. Users should verify the information from reliable sources before relying on ChatGPT responses. ChatGPT utilises just raw text devoid of any links or citations. Unlike Google, it is difficult to confirm the correctness of its responses. In addition, Google is also developing sizable language models of its own and heavily using AI in its search engines as ChatGPT advances. ChatGPT, a sizable language model, is continually trained to improve answer accuracy. Nevertheless, since it is a new technology, the model still needs to receive more instruction. As a result, the AI chatbot could provide inaccurate information. ChatGPT's training data has limits, much like many AI models. The bias in the data and the limitations in the training data might have a detrimental effect on the model's output.

Of course, these new AI technologies cannot read minds. To produce the outcomes the human user is looking for, a novel but less complex kind of human creativity is required in the form of text prompts. The AI system creates consecutive rounds of outputs using iterative prompting, an example of human-AI cooperation until the person authoring the prompts is pleased with the outcomes. The specific risks and possible consequences connected with ChatGPT will ultimately determine the necessary amount of regulation. As with any powerful new technology, it is crucial to carefully assess its possible effects and take precautions to ensure it is utilised morally and responsibly. While many people have praised ChatGPT for increasing their productivity, others have expressed worry about it for understandable reasons. Schools, colleges, and education boards have expressed concerns about employing this technology for submissions and examinations. Not to add that Stack Overflow, a platform for developers, prohibited the usage of chatbots immediately after the launch of ChatGPT due to increased results inaccuracy.

8. Future scope

ChatGPT is picking up new prompts as more and more users feed it. By forgoing conventional education in favour of ChatGPT's shortcuts, students improve AI, making it more useful for users in the future. ChatGPT will have a significant influence on the educational technology sector as well. Many edtech businesses may now provide a subject's foundations while using ChatGPT to give students a place to ask questions and have their worries cleared. Despite its shortcomings, ChatGPT will prove helpful in real-world use situations.

Consequently, companies are eager to use it for profit-making objectives. In the future, AI can help identify which students would profit

the most from which tutors, those who cannot only help bridge learning gaps but also serve as relevant sources of mentorship, guidance, and inspiration. Designing new AI applications needs careful consideration, particularly in light of our society's reactions to the rapidly evolving AI environment, which is a complex mix of fear, optimism, anxiety, astonishment, and wonder.

9. Conclusion

ChatGPT has provided remarkable success since its debut in November 2022. It can generate essays, fictitious tales, haikus, and even cover letters for job applications. ChatGPT can provide solutions to life's most significant and most minor problems. It does this with the help of meticulous supervision from human specialists and information gathered from an incredible volume of material on the internet. ChatGPT can conduct human-like talks on a variety of subjects using natural language. Users of ChatGPT are utilising the platform to help with composing emails, programming code, and answering inquiries on a variety of subjects, including investing. ChatGPT has received extremely excellent feedback thus far, with many appreciating its sophisticated features and simplicity of its use. ChatGPT has the potential to be a significant player in natural language processing. The chatbot responds to our inquiries in a conversational, albeit slightly stiff, manner using the platform of OpenAI. Particularly noteworthy has been ChatGPT's capacity to comprehend and react to a wide variety of issues; some have even suggested that it may completely alter how humans engage with technology. In future, ChatGPT's features will be an excellent tool for businesses in industries like customer service, online learning, and market research. OpenAI and its most significant investors have invested billions in developing, training, and using these models. It may be a wise investment in the long term, positioning OpenAI at the forefront of AI creative tools.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] M. Dowling, B. Lucey, ChatGPT for (finance) research: The Bananarama conjecture, Finance Res. Lett. (2023) 103662.

[2] A. Gilson, C.W. Safranek, T. Huang, V. Socrates, L. Chi, R.A. Taylor, D. Chartash, How does chatgpt perform on the united states medical licensing examination? The implications of large language models for medical education and knowledge assessment, JMIR Med. Educ. 9 (1) (2023) e45312.

[3] J. Rudolph, S. Tan, S. Tan, ChatGPT: Bullshit spewer or the end of traditional assessments in higher education? J. Appl. Learn. Teach. 6 (1) (2023).

[4] B.D. Lund, T. Wang, Chatting about ChatGPT: how may AI and GPT impact academia and libraries? Library Hi Tech News, 2023.

[5] T.H. Kung, M. Cheatham, A. Medenilla, C. Sillos, L. De Leon, C. Elepaño, et al., Performance of ChatGPT on USMLE: Potential for AI-assisted medical education using large language models, PLOS Digit. Health 2 (2) (2023) e0000198.

[6] Y. Shen, L. Heacock, J. Elias, K.D. Hentel, B. Reig, G. Shih, L. Moy, ChatGPT and other large language models are double-edged swords, Radiology (2023) 230163.

[7] E.A. van Dis, J. Bollen, W. Zuidema, R. van Rooij, C.L. Bockting, ChatGPT: Five priorities for research, Nature 614 (7947) (2023) 224–226.

[8] M. Liebrenz, R. Schleifer, A. Buadze, D. Bhugra, A. Smith, Generating scholarly content with ChatGPT: Ethical challenges for medical publishing, Lancet Digit. Health (2023).

[9] Ö. Aydın, E. Karaarslan, OpenAI ChatGPT generated literature review: Digital twin in healthcare, 2022, Available At SSRN 4308687.

[10] Chen T.J., ChatGPT and other artificial intelligence applications speed up scientific writing, J. Chin. Med. Assoc. 1 (2023) 0–1097.

[11] H. Alkaissi, S.I. McFarlane, Artificial hallucinations in ChatGPT: Implications in scientific writing, Cureus 15 (2) (2023).

[12] L. Ante, E. Demir, The ChatGPT effect on AI-themed cryptocurrencies, 2023, Available At SSRN 4350557.

[13] D. Mhlanga, Open AI in education, the responsible and ethical use of ChatGPT towards lifelong learning, in: Education, the Responsible and Ethical Use of ChatGPT Towards Lifelong Learning (February 11, 2023), 2023.

[14] L. De Angelis, F. Baglivo, G. Arzilli, G.P. Privitera, P. Ferragina, A.E. Tozzi, C. Rizzo, ChatGPT and the rise of large language models: The new AI-driven infodemic threat in public health, 2023, Available At SSRN 4352931.

[15] H. Donato, P. Escada, T. Villanueva, The transparency of science with chatgpt and the emerging artificial intelligence language models: Where should medical journals stand? Acta MÉ (2023).

[16] S. Huh, Are ChatGPT's knowledge and interpretation ability comparable to those of medical students in Korea for taking a parasitology examination?: A descriptive study, J. Educ. Eval. Health Prof. 20 (1) (2023).

[17] B. Kutela, K. Msechu, S. Das, E. Kidando, Chatgpt's scientific writings: A case study on traffic safety, 2023, Available At SSRN 4329120.

[18] S. Biswas, ChatGPT and the future of medical writing, Radiology (2023) 223312.

[19] A.H. Kumar, Analysis of ChatGPT tool to assess the potential of its utility for academic writing in biomedical domain, Biol. Eng. Med. Sci. Rep. 9 (1) (2023) 24–30.

[20] D. Street, J. Wilck, 'Let's have a chat': Principles for the effective application of ChatGPT and large language models in the practice of forensic accounting, 2023, Available At SSRN 4351817.

[21] S. Pal, Performing effective research using ChatGPT, Indian J. Comput. Sci. 7 (6) (2022) 8–15.

[22] C. Zielinski, M. Winker, R. Aggarwal, L. Ferris, M. Heinemann, J.F. Lapeña, et al., Chatbots, ChatGPT, and scholarly manuscripts WAME recommendations on ChatGPT and chatbots in relation to scholarly publications, Afro-Egypt. J. Infect. Endemic Dis. (2023).

[23] A.S. George, A.H. George, A review of ChatGPT AI's impact on several business sectors, Partn. Univers. Int. Innov. J. 1 (1) (2023) 9–23.

[24] J. Deng, Y. Lin, The benefits and challenges of ChatGPT: An overview, Front. Comput. Intell. Syst. 2 (2) (2022) 81–83.

[25] S. Hargreaves, 'Words are Flowing Out Like Endless Rain Into a Paper Cup': ChatGPT & Law School Assessments, The Chinese University of Hong Kong Faculty of Law Research Paper, 2023, 2023-03.

[26] T. Sakirin, R.B. Said, User preferences for ChatGPT-powered conversational interfaces versus traditional methods, Mesop. J. Comput. Sci. 2022 (2022) 5–12.

[27] D.L. Mann, Artificial intelligence discusses the role of artificial intelligence in translational medicine: A JACC: Basic to translational science interview with ChatGPT, Basic Transl. Sci. (2023).

[28] N. Anderson, D.L. Belavy, S.M. Perle, S. Hendricks, L. Hespanhol, E. Verhagen, A.R. Memon, AI did not write this manuscript, or did it? Can we trick the AI text detector into generated texts? The potential future of ChatGPT and AI in sports & exercise medicine manuscript generation, BMJ Open Sport Exerc. Med. 9 (1) (2023) e001568.

[29] M. M Alshater, Exploring the role of artificial intelligence in enhancing academic performance: A case study of ChatGPT, 2022, Available At SSRN.

[30] A.M. Perlman, The implications of OpenAI's assistant for legal services and society, 2022, Available At SSRN.

[31] N. Helberger, N. Diakopoulos, ChatGPT and the AI act, Internet Policy Rev. 12 (1) (2023).

[32] M. Mijwil, M. Aljanabi, Towards artificial intelligence-based cybersecurity: The practices and ChatGPT generated ways to combat cybercrime, Iraqi J. Comput. Sci. Math. 4 (1) (2023) 65–70.

[33] B. Gordijn, H.T. Have, ChatGPT: Evolution or revolution? Med. Health Care Philos. (2023) 1–2.

[34] L. Floridi, AI as agency without intelligence: On ChatGPT large language models, and other generative models, Philos. Technol. (2023).

[35] J.V. Pavlik, Collaborating with ChatGPT: Considering the implications of generative artificial intelligence for journalism and media education, Journalism Mass Commun. Educ. (2023) 10776958221149577.

[36] C.A. Gao, F.M. Howard, N.S. Markov, E.C. Dyer, S. Ramesh, Y. Luo, A.T. Pearson, Comparing scientific abstracts generated by ChatGPT to original abstracts using an artificial intelligence output detector, plagiarism detector, and blinded human reviewers, 2022, BioRxiv, 2022-12.

[37] M. Aljanabi, M. Ghazi, A.H. Ali, S.A. Abed, ChatGpt: Open possibilities, Iraqi J. Comput. Sci. Math. 4 (1) (2023) 62–64.

[38] M. Mijwil, M. Aljanabi, A.H. Ali, ChatGPT: Exploring the role of cybersecurity in the protection of medical information, Mesop. J. CyberSecur. 2023 (2023) 18–21.

[39] D. Baidoo-Anu, L. Owusu Ansah, Education in the Era of generative artificial intelligence (AI): Understanding the potential benefits of ChatGPT in promoting teaching and learning, 2023, Available At SSRN 4337484.

[40] X. Zhai, ChatGPT user experience: Implications for education, 2022, Available At SSRN 4312418.

[41] L. Bishop, A computer wrote this paper: What ChatGPT means for education, research, and writing, in: Research, and Writing (January 26, 2023), 2023.

[42] V. Taecharungroj, What can ChatGPT do? Analysing early reactions to the innovative AI chatbot on Twitter, Big Data Cogn. Comput. 7 (1) (2023) 35.

[43] M. Aljanabi, ChatGPT: Future directions and open possibilities, Mesop. J. CyberSecur. 2023 (2023) 16–17.

[44] C. Macdonald, D. Adeloye, A. Sheikh, I. Rudan, Can ChatGPT draft a research article? An example of population-level vaccine effectiveness analysis, J. Glob. Health (13) (2023).

[45] R.J.M. Ventayen, OpenAI ChatGPT generated results: Similarity index of artificial intelligence-based contents, 2023, Available At SSRN 4332664.
[46] H. Alshurafat, The usefulness and challenges of chatbots for accounting professionals: Application on ChatGPT, 2023, Available At SSRN 4345921.
[47] M.R. King, chatGPT, A conversation on artificial intelligence, chatbots, and plagiarism in higher education, Cell. Mol. Bioeng. (2023) 1–2.
[48] R.S. D'Amico, T.G. White, H.A. Shah, D.J. Langer, I asked a ChatGPT to write an editorial about how we can incorporate chatbots into neurosurgical research and patient care..., Neurosurgery 1 (2022) 0–1227.
[49] X. Zhai, ChatGPT for next generation science learning, 2023, Available At SSRN 4331313.
[50] A. Zarifhonarvar, Economics of ChatGPT: A labor market view on the occupational impact of artificial intelligence, 2023, Available At SSRN 4350925.
[51] B. Lund, D. Agbaji, Information literacy, data literacy, privacy literacy, and ChatGPT: Technology literacies align with perspectives on emerging technology adoption within communities, in: Data Literacy, Privacy Literacy, and ChatGPT: Technology Literacies Align with Perspectives on Emerging Technology Adoption Within Communities (January 14, 2023), 2023.
[52] A. Flanagin, K. Bibbins-Domingo, M. Berkwits, S.L. Christiansen, Nonhuman authors and implications for the integrity of scientific publication and medical knowledge, JAMA (2023).
[53] C. Stokel-Walker, R. Van Noorden, What ChatGPT and generative AI mean for science, Nature 614 (7947) (2023) 214–216.
[54] W. Geerling, G.D. Mateer, J. Wooten, N. Damodaran, Is ChatGPT smarter than a student in principles of economics? 2023, Available At SSRN 4356034.
[55] L. Avila-Chauvet, D. Mejía, C.O. Acosta Quiroz, Chatgpt as a support tool for online behavioral task programming, 2023, Available At SSRN 4329020.
[56] F. Ali, Let the devil speak for itself: Should ChatGPT be allowed or banned in hospitality and tourism schools? J. Glob. Hospit. Tourism 2 (1) (2023) 1–6.
[57] O. Oviedo-Trespalacios, A.E. Peden, T. Cole-Hunter, A. Costantini, M. Haghani, S. Kelly, et al., The risks of using ChatGPT to obtain common safety-related information and advice, 2023, Available At SSRN 4346827.
[58] T. Yue, D. Au, C.C. Au, K.Y. Iu, Democratising financial knowledge with ChatGPT by OpenAI: Unleashing the power of technology, 2023, Available At SSRN 4346152.
[59] S. Nisar, M.S. Aslam, Is ChatGPT a good tool for T & CM students in studying pharmacology? 2023, Available At SSRN 4324310.
[60] R. Macey-Dare, ChatGPT & generative AI systems as quasi-expert legal advice lawyers-case study considering potential appeal against conviction of tom hayes, 2023, Available At SSRN 4342686.
[61] T. Hirosawa, Y. Harada, M. Yokose, T. Sakamoto, R. Kawamura, T. Shimizu, Diagnostic accuracy of differential-diagnosis lists generated by generative pretrained transformer 3 chatbot for clinical Vignettes with common chief complaints: A pilot study, Int. J. Environ. Res. Public Health 20 (4) (2023) 3378.

TBench Editorial Board

More information about this series at https://www.benchcouncil.org/tbench/

TBench Call For Papers

BenchCouncil Transactions on Benchmarks, Standards and Evaluations (TBench)
ISSN:2772-4859

Aims and Scopes

BenchCouncil Transactions on Benchmarks, Standards, and Evaluations (TBench) publishes position articles that open new research areas, research articles that address new problems, methodologies, tools, survey articles that build up comprehensive knowledge, and comments articles that argue the published articles. The submissions should deal with the benchmarks, standards, and evaluation research areas. Particular areas of interest include, but are not limited to:

- 1. Generalized benchmark science and engineering (see https://www.sciencedirect.com/science/article/pii/S2772485921000120), including but not limited to
  - measurement standards
  - standardized data sets with defined properties
  - representative workloads
  - representative data sets
  - best practices
- 2. Benchmark and standard specifications, implementations, and validations of:
  - Big Data
  - AI
  - HPC
  - Machine learning
  - Big scientific data
  - Datacenter
  - Cloud
  - Warehouse-scale computing
  - Mobile robotics
  - Edge and fog computing
  - IoT
  - Chain block
  - Data management and storage
  - Financial domains
  - Education domains
  - Medical domains
  - Other application domains
- 3. Data sets
  - Detailed descriptions of research or industry datasets, including the methods used to collect the data and technical analyses supporting the quality of the measurements.
  - Analyses or meta-analyses of existing data and original articles on systems, technologies, and techniques that advance data sharing and reuse to support reproducible research.
  - Evaluating the rigor and quality of the experiments used to generate the data and the completeness of the data description.
  - Tools generating large-scale data while preserving their original characteristics.
- 4. Workload characterization, quantitative measurement, design, and evaluation studies of:
  - Computer and communication networks, protocols, and algorithms
  - Wireless, mobile, ad-hoc and sensor networks, IoT applications
  - Computer architectures, hardware accelerators, multi-core processors, memory systems, and storage networks
  - High-Performance Computing
  - Operating systems, file systems, and databases

- Virtualization, data centers, distributed and cloud computing, fog, and edge computing
- Mobile and personal computing systems
- Energy-efficient computing systems
- Real-time and fault-tolerant systems
- Security and privacy of computing and networked systems
- Software systems and services, and enterprise applications
- Social networks, multimedia systems, Web services
- Cyber-physical systems, including the smart grid

- 5. Methodologies, metrics, abstractions, algorithms, and tools for:
  - Analytical modeling techniques and model validation
  - Workload characterization and benchmarking
  - Performance, scalability, power, and reliability analysis
  - Sustainability analysis and power management
  - System measurement, performance monitoring, and forecasting
  - Anomaly detection, problem diagnosis, and troubleshooting
  - Capacity planning, resource allocation, run time management, and scheduling
  - Experimental design, statistical analysis, simulation
- 6. Measurement and evaluation
  - Evaluation methodology and metric
  - Testbed methodologies and systems
  - Instrumentation, sampling, tracing, and profiling of Large-scale real-world applications and systems
  - Collection and analysis of measurement data that yield new insights
  - Measurement-based modeling (e.g., workloads, scaling behavior, assessment of performance bottlenecks)
  - Methods and tools to monitor and visualize measurement and evaluation data
  - Systems and algorithms that build on measurement-based findings
  - Advances in data collection, analysis, and storage (e.g., anonymization, querying, sharing)
  - Reappraisal of previous empirical measurements and measurement-based conclusions
  - Descriptions of challenges and future directions the measurement and evaluation community should pursue

Bench 2022 Call For Papers

2022 BenchCouncil International Symposium on Benchmarking, Measuring and Optimizing (Bench 2022) Calls For Papers

https://www.benchcouncil.org/bench22/index.html

Full Papers deadline: July 28, 2022, 23:59:59 AoE
Notification: September 6, 2022, 23:59:59 AoE
Final Papers due: October 11, 2022, 23:59:59 AoE
Conference date: Nov. 7th - Nov. 9th, 2022 (Virtual)
Submission site: https://bench2022.hotcrp.com/

Introduction

Benchmarks, Data, Standards, Measurements, and Optimizations are fundamental human activities and assets. The Bench conference has two essential duties: promote data or benchmark-based quantitative approaches to tackle multidisciplinary and interdisciplinary challenges; connect architecture, system, data management, algorithm, and application communities to better co-design for the inherent workload characterizations.

The Bench conference provides a high-quality, single-track forum for presenting results and discussing ideas that further the knowledge and understanding of the benchmarks, data, standards, measurements, and optimizations community as a whole. It is a multidisciplinary and interdisciplinary conference. The past meetings attracted researchers and practitioners from the architecture, system, algorithm, and application communities. It includes both invited sessions and contributed sessions.

Regularly, the Bench conference will present the BenchCouncil Achievement Award ($3000), the BenchCouncil Rising Star Award ($1000), the BenchCouncil Best Paper Award ($1000), and the BenchCouncil Distinguished Doctoral Dissertation Awards in Computer Architecture ($1000) and in other areas ($1000). This year, the BenchCouncil Distinguished Doctoral Dissertation Award includes two tracks: computer architecture and other areas. Among the submissions of each track, four candidates will be selected as finalists. They will be invited to give a 30-minute presentation at the Bench’22 Conference and contribute research articles to BenchCouncil Transactions on Benchmarks, Standards and Evaluation. Finally, for each track, one among the four will receive the award for each track, which carries a $1,000 honorarium.

Organization

General Co-Chairs

Emmanuel Jeannot, INRIA, France
Peter Mattson, Google, USA
Wanling Gao, University of Chinese Academy of Sciences, China

Program Co-Chairs

Chunjie Luo, ICT, Chinese Academy of Sciences, China
Ce Zhang, ETH Zurich, Switzerland
Ana Gainaru, Oak Ridge National Laboratory, USA

Publicity Co-Chairs

David Kanter, MLCommons
Rui Ren, Beijing Institute of Open Source Chip
Zhen Jia, Amazon

Web Co-Chairs

Jiahui Dai, BenchCouncil

Jiahui Dai, BenchCouncil
Qian He, Beijing Institute of Open Source Chip

Award Committees

BenchCouncil Distinguished Doctoral Dissertation Award Committee in Other Areas:
Jack Dongarra, University of Tennessee
Xiaoyi Lu, The University of California, Merced
Jeyan Thiyagalingam, STFC-RAL
Lei Wang, ICT, Chinese Academy of Sciences
Spyros Blanas, The Ohio State University

BenchCouncil Distinguished Doctoral Dissertation Award Committee in Computer Architecture:
Peter Mattson, Google
Vijay Janapa Reddi, Harvard University
Wanling Gao, Chinese Academy of Sciences

Bench Steering Committees

Jack Dongarra, University of Tennessee
Geoffrey Fox, Indiana University
D. K. Panda, The Ohio State University
Felix, Wolf, TU Darmstadt
Xiaoyi Lu, University of California, Merced
Resit Sendag, University of Rhode Island, USA
Wanling Gao, ICT, Chinese Academy of Sciences & UCAS
Jianfeng Zhan, ICT, Chinese Academy of Sciences &BenchCouncil

Call For Papers

The Bench conference encompasses a wide range of areas and topics in benchmarking, measurement, evaluation methods and tools. We solicit papers describing original and previously unpublished work. The areas and topics of interest include, but are not limited to the following.

- 1. Areas:
  - Architecture
  - Data Management
  - Algorithm
  - Datasets
  - System
  - Network
  - Reliability and Security
  - Application
- 2. Topics:
  - Benchmark and standard specifications, implementations, and validations
  - Dataset Generation and Analysis
  - Workload characterization, quantitative measurement, design and evaluation studies
  - Methodologies, abstractions, metrics, algorithms and tools
  - Measurement and evaluation

Paper Submission

Papers must be submitted in PDF. For a full paper, the page limit is 15 pages in the LNCS format, not including references. For a short paper, the page limit is 8 pages in the LNCS format, not including references. The submissions will be judged based on the merit of the ideas rather than the length. The reviewing process is double-blind. Upon acceptance, the proceeding will be published by Springer LNCS (Indexed by EI). Please note that the LNCS format is the final one for publishing. Distinguished papers will be recommended to and published by the BenchCouncil Transactions on Benchmarks, Standards and Evaluation (TBench).
At least one author must pre-register for the symposium, and at least one author must attend the symposium to present the paper. Papers for which no author is pre-registered will be removed from the proceedings.

Submission site:https://bench2022.hotcrp.com/
LNCS Latex template:https://www.benchcouncil.org/file/llncs2e.zip

Awards

* BenchCouncil Achievement Award ($3,000)
- This award recognizes a senior member who has made long-term contributions to benchmarking, measuring, and optimizing. The winner is eligible for the status of a BenchCouncil Fellow.

* BenchCouncil Rising Star Award ($1,000)
- This award recognizes a junior member who demonstrates outstanding potential for research and practice in benchmarking, measuring, and optimizing.

* BenchCouncil Best Paper Award ($1,000)
- This award recognizes a paper presented at the Bench conferences, which demonstrates potential impact on research and practice in benchmarking, measuring, and optimizing.

* BenchCouncil Distinguished Doctoral Dissertation Award ($2000)
- This award recognizes and encourages superior research and writing by doctoral candidates in the broad field of benchmarks, data, standards, evaluations, and optimizations community. This year, the award includes two tracks, including the BenchCouncil Distinguished Doctoral Dissertation Award in Computer Architecture ($1000) and BenchCouncil Distinguished Doctoral Dissertation Award in other areas ($1000).

Technical Program Committee

Murali Krishna Emani, ANL
Shin-ying Lee, AMD
Steve Farrell, NERSC
Krishnakumar Nair, Meta
Greg Diamos, Landing.AI
Fei Sun, Alibaba
Narayanan Sundaram, Facebook
Zhen Jia, Amazon
Shengen Yan, SenseTime
Gang Lu, Tencent
Rui Ren, Beijing Open-Source IC Academy
Bin Hu, ICT, CAS
Khaled Ibrahim, Lawrence Berkeley National Laboratory
Sascha Hunold, TU Wien
Woongki Baek, UNIST
Mario Marino, Leeds Beckett University
Bin Ren, William & Mary

Gwangsun Kim, POSTECH
Vladimir Getov, University of Westminster
Guangli Li, ICT, CAS
Biwei Xie, ICT, CAS
Nicolas Rougier, INRIA