BenchCouncil Transactions on Benchmarks, Standards and Evaluations

TBench
Volume 3, Issue 2
2023

Original Articles

- Unlocking the opportunities through ChatGPT Tool towards ameliorating the education system
  Mohd Javaid, Abid Haleem, Ravi Pratap Singh, Shahbaz Khan, Ibrahim Haleem Khan

- Benchmarking HTAP databases for performance isolation and real-time analytics
  Guoxin Kang, Simin Chen, Hongxiao Li

- CoviDetector: A transfer learning-based semi supervised approach to detect Covid-19 using CXR images
  Deepraj Chowdhury, Anik Das, Ajoy Dey, Soham Banerjee, ... Steve Uhlig

- DPUBench: An application-driven scalable benchmark suite for comprehensive DPU evaluation
  Zheng Wang, Chenxi Wang, Lei Wang

- StreamAD: A cloud platform metrics-oriented benchmark for unsupervised online anomaly detection
  Jiahui Xu, Chengxiang Lin, Fengrui Liu, Yang Wang, ... Gaogang Xie

ISSN: 2772-4859

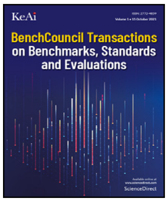
BenchCouncil Transactions on Benchmarks, Standards and Evaluations (TBench) is an open-access multi-disciplinary journal dedicated to benchmarks, standards, evaluations, optimizations, and data sets. This journal is a peer-reviewed, subsidized open access journal where The International Open Benchmark 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

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/



Full length article

Unlocking the opportunities through ChatGPT Tool towards ameliorating the education system

Mohd Javaid [a,*], Abid Haleem [a], Ravi Pratap Singh [b], Shahbaz Khan [c], Ibrahim Haleem Khan [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] *College of Engineering, Northeastern University, Boston, MA, USA*

ARTICLE INFO

Keywords:
Artificial Intelligence
ChatGPT
Education
Students
Teaching
Learning

ABSTRACT

Artificial Intelligence (AI)-based ChatGPT developed by OpenAI is now widely accepted in several fields, including education. Students can learn about ideas and theories by using this technology while generating content with it. ChatGPT is built on State of the Art (SOA), like Deep Learning (DL), Natural Language Processing (NLP), and Machine Learning (ML), an extrapolation of a class of ML-NLP models known as Large Language Model (LLMs). It may be used to automate test and assignment grading, giving instructors more time to concentrate on instruction. This technology can be utilised to customise learning for kids, enabling them to focus more intently on the subject matter and critical thinking ChatGPT is an excellent tool for language lessons since it can translate text from one language to another. It may provide lists of vocabulary terms and meanings, assisting students in developing their language proficiency with resources. Personalised learning opportunities are one of ChatGPT's significant applications in the classroom. This might include creating educational resources and content tailored to a student's unique interests, skills, and learning goals. This paper discusses the need for ChatGPT and the significant features of ChatGPT in the education system. Further, it identifies and discusses the significant applications of ChatGPT in education. Using ChatGPT, educators may design lessons and instructional materials specific to each student's requirements and skills based on current trends. Students may work at their speed and concentrate on the areas where they need the most support, resulting in a more effective and efficient learning environment. Both instructors and students may profit significantly from using ChatGPT in the classroom. Instructors may save time on numerous duties by using this technology. In future, ChatGPT will become a powerful tool for enhancing students' and teachers' experience.

1. Introduction

ChatGPT is a revolutionary tool that responds to inquiries on nearly anything available in the contemporary digital environment to the dataset it has been trained. Now, ChatGPT is innovative in generating logical, cohesive, pertinent, and fluent replies, giving the sense that someone is physically typing what we see on the screen. In education, instructors may use ChatGPT in their courses and utilise it to tailor the learning experience for their students. On the other hand, students' writing abilities may be enhanced by using text completion, translation, and text summarising tools. ChatGPT's capabilities may be used to identify content bias and fix problems with educational materials. Given the growing need for updated teaching materials, ChatGPT can assist the state in creating and implementing an impartial and fair curriculum. If implemented appropriately, this might act as a bridge to lessen the pressure on a stressed-out educational system [1–3].

ChatGPT is an effective tool for instructors to improve their lessons and students' learning. It will not replace teachers. Instead, make them more powerful with better hands-on resources. Teachers may help their students learn more effectively by utilising ChatGPT to stimulate conversations, provide tailored feedback, and improve their language and literacy abilities. Individual students may get tailored feedback and coaching using ChatGPT [4,5]. The application may provide detailed comments on a student's writing project, offering recommendations for development and motivation. Students may feel more self-assured and inspired to keep studying and developing. ChatGPT generates a response by reading a text, such as a phrase or a prompt, and then understanding the problem statement. Given the context of the words before it, the model is trained to predict the next word in a phrase.

ChatGPT may be used to grade essays automatically with reasoning and even better solutions. With the help of this function, instructors

* 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), haleemkhan.i@northeastern.edu (I.H. Khan).

https://doi.org/10.1016/j.tbench.2023.100115
Received 13 April 2023; Received in revised form 20 May 2023; Accepted 20 May 2023
Available online 26 May 2023

may mark written assignments and provide comments on grammar, structure, plagiarism and content. Producing ideas, summaries, and even whole talks, may also aid with the composition of speeches. It may aid with research by guiding students in locating and organising data for papers and other types of study. ChatGPT may provide students learning a language immediate feedback on their pronunciation and grammar, assisting them in swiftly and effectively developing their language abilities. It may help students with trouble reading and writing by suggesting ways to improve their phrases and paragraphs [6,7].

The ChatGPT language model has the power to create writing that is similar to what a person would write. It can perform various natural language processing tasks, including language translation, text summarisation, text creation, and conversation systems. It was trained on a large dataset of online content such as webpages, research articles, books, social media posts and chatter. ChatGPT typically performs best when conversing in human language, remembering previous exchanges within the same conversation, referring to physical, emotional, and cultural experiences in the training data, and dynamically drawing from a scientific and technical knowledge pool to address queries [8–10]. ChatGPT can produce language comparable to how people write by training on such a broad dataset.

ChatGPT and other big language models' capacity for content development may aid marketers in becoming more productive and successful. This enables marketers to scale up content personalisation, which was previously time-consuming. It has long been used in various ways, including conversational chatbots, automation, and data analysis. Today's teachers can think about how ChatGPT might act as a writing tutor for their students. Students might use this tool to quickly assess their writing without waiting for an instructor's response. The students could then request specific actions from the AI tool to be taken to edit or revise their work. In terms of creative content, it functions as a super-effective word organiser [11,12]. The main aim of this paper is to discuss the significant applications of ChatGPT in education.

2. ChatGPT

An AI-powered chatbot is called ChatGPT by OpenAI. The term "Generative Pre-trained Transformer (GPT)" refers to a language processing model trained on massive data to produce writing that resembles a person's. ChatGPT is a natural language processing technology that uses AI to respond to quarries. As a result, it creates information more conversationally, picks up knowledge from those talks, and then can provide ever more specifically customised replies. ChatGPT behaves like a person while giving instructions to users and providing information. This technology can do various activities, including creating poetry, coding, answering inquiries, writing emails and essays, and translating documents. ChatGPT differs from other chatbots in that it can respond instantly, resulting in more varied and lively discussions on almost all topics [13–15].

ChatGPT is discussed as a tool for improving students' skills by fostering their ability to ask questions and formulate them precisely, expanding their knowledge through ChatGPT's answers, and teaching skills to assess the accuracy, reliability, and quality of ChatGPT's answers as well as to filter the pertinent information gleaned from answers. This technology suits various applications since it can adjust to different conditions and situations. ChatGPT is a flexible tool that may be used in various natural language processing applications. It can respond to instructions with high accuracy and fluency, but it needs a thorough understanding of the world and the ability to think like a person [16,17].

ChatGPT is an AI-based tool for having exciting and genuine talks with people. It can comprehend text-based input and react to it using AI and ML methods. As a result, it can have discussions that are more comparable to those between two people. As a result, it is an effective tool for businesses in the customer service, marketing, and content development sectors. ChatGPT functions as a virtual assistant that can converse with us and respond to our inquiries like an actual person would [18,19]. ChatGPT uses deep learning algorithms to generate human–machine natural language discussions.

3. Need of ChatGPT in education

ChatGPT can affect several aspects of education, including writing, instruction method and teaching pedagogy. Writing has been essential to fostering creative and critical thinking for ages through organising information and creating narratives. It continues to play a crucial role in education, even in the age of AI. Therefore, we should concentrate on offering insights that are incomprehensible to AI. Students' thesis, assignments, and essay writing should be condensed, reflective, and grounded in a particular setting. Education and AI are essential topics in conversations about our society's future. ChatGPT is a valuable tool for writers, marketers, and other professionals that often need to produce text. This has several uses, including producing content for websites, social networking platforms, marketing materials, and chatbot replies [20–22].

Integrating ChatGPT into higher education might result in a shift towards AI, diminishing the need for professors and possibly lowering opportunities for interpersonal relationships and human engagement. In order to assist students and improve their writing abilities, ChatGPT may check for grammatical and structural problems in their work and provide valuable comments. In order to understand and concentrate on the areas they need to improve, students can also receive personalised feedback based on their writing style. Computers may imitate human conversations using ChatGPT [23,24]. This can accurately respond to user inquiries and personalisation by comprehending user intent and context [25,26]. The students could explore several things with the help of ChatGPT, such as developing a computer program, writing an essay and solving a mathematical problem. All these things could be possible with ChatGPT.

ChatGPT is designed to connect quickly and respond more cohesively, like conversational tools like bots and virtual assistants. Because of this interoperability, businesses may expand on their current offerings and develop distinctive AI-powered chatbots that can quickly comprehend and respond to consumer demands. Conversations may be held in a private and secure environment using ChatGPT. It offers a secure environment free from intervention or manipulation by using AI to identify harmful information, spam, and censorship. Moreover, ChatGPT never stores nor sends personal information to other parties. Thus, to preserve users' privacy, all correspondence is encrypted and kept locally [27,28].

4. Research objectives

ChatGPT is an AI-driven natural language processing application enabling users to participate in human-like text-based discussions with AI-based software. It may provide information, help write essays and letters, and produce code and websites. ChatGPT has the potential to complete transformation the way we teach and learn. Teachers may provide students with immediate feedback and aid their knowledge growth by using ChatGPT in the classroom. ChatGPT may be used to automate monotonous routines and save up instructors' time so they can concentrate on teaching more insightful courses. ChatGPT can change how we interact with chatbots entirely. ChatGPT can understand natural language, interpret context, and generate responses to engage in fruitful conversations with people by utilising the power of AI. Moreover, ChatGPT may be used to assist students in preparing for debates by producing plausible arguments and counterarguments on a particular subject. It may provide innovative writing ideas to motivate students and assist them in enhancing their writing abilities [29–31]. The primary research objectives of this article are as under:

RO1: - to identify what needs of education can be fulfilled by ChatGPT;

RO2: - to study the significant features of ChatGPT towards the education system;

RO3: - to study the workflow elements of ChatGPT for the education system;

RO4: - to identify and discuss the significant applications of ChatGPT in education;

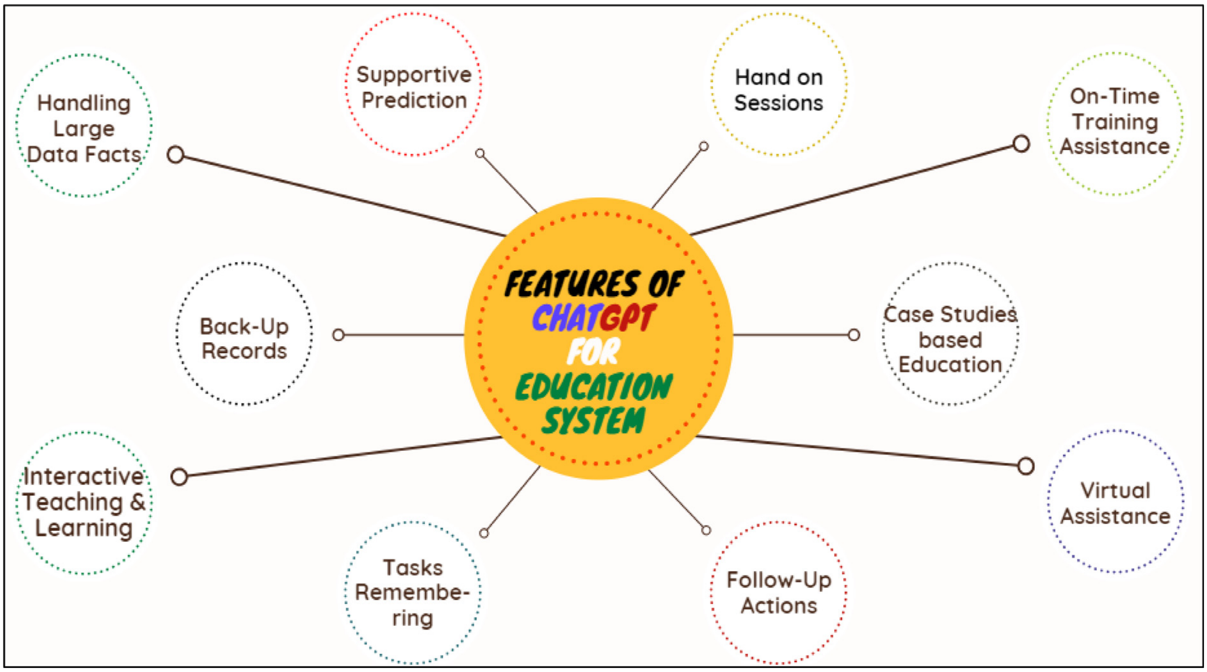

FEATURES OF CHATGPT FOR EDUCATION SYSTEM
Handling Large Data Facts
Supportive Prediction
Hand on Sessions
On-Time Training Assistance
Back-Up Records
Case Studies based Education
Interactive Teaching & Learning
Tasks Remembe-ring
Follow-Up Actions
Virtual Assistance

Fig. 1. Influential capabilities of ChatGPT for education system.

5. Significant features of ChatGPT towards education system

Fig. 1 explores the various associated typical capabilities, features, and applications of ChatGPT support for education. It includes the features like remembering aspects, prediction support, translation creation, etc. [32,33]. In addition to this, several associated other characteristics and classical perspectives of ChatGPT are further represented and elaborated in Fig. 1.

In education, new technology constantly emerges and often vanishes over time, while only some innovations are ingrained in the system. ChatGPT generates a fantastic set of issues for students to work together on, and of course, students may build these problems independently. Even though they may have learned about probability via experience, activities like this solidify their understanding of the subject by encouraging students to work together on various approaches, test their theories, and refine them. These kinds of problems may be quickly created when a gap is found. ChatGPT enables teachers and students to create various materials, including writing prompts, discussion topics, puzzles, and much more [24,33]. Learners may produce these as needed, and they can also self-direct their research and practice.

ChatGPT provides a step-by-step explanation, which includes visual examples and common pitfalls and is far superior to Google's response. The irruption of ChatGPT shocked educational institutions around the globe once again. ChatGPT would be both the best instructor and the best student. In addition, with the help of technologies like AI, instructors and students may increase their powers and opportunities, just like they did with maths calculators in the past. An AI chatbot may be hired to provide students with rapid answers to frequently requested topics, much like a learning assistant with reasoning. This support may be helpful as students continue their education outside and after the lesson. By enhancing search and offering individualised suggestions on material and other learning resources, AI-powered learning assistants may be utilised to direct and help students with their learning [34,35].

One of the numerous ways ChatGPT might be utilised in the classroom is to create outlines. It could come up with lesson plans personalised to each student and come up with suggestions for class projects. It might be used as a debate partner or an after-hours tutor. It might serve as the basis for class exercises or as a tool to help English language learners develop their fundamental writing abilities. Unstructured data is a challenge in the age of the digital revolution. The issue is that they need to be more challenging to manage, arrange, sort and analyse [36,37]. ChatGPT is helpful since it can convert unstructured data into structured data. By offering clarifications, recommendations, and examples, ChatGPT may help in locating and resolving coding issues. ChatGPT may be used to target specific people with the content. Businesses may use the model to generate personalised content like emails, social media posts, and product recommendations by training the model on a collection of user data [38,39].

With a sophisticated language model, ChatGPT has the potential to alter the way we work and learn thoroughly. Providing with the information in seconds instead of It is a helpful tool for professionals, educators, and students since it can produce text that looks and sounds like human speech. The potential of ChatGPT is limitless, given the ongoing research and breakthroughs in natural language processing. ChatGPT enables users to conduct virtually human-like dialogues to address issues as diverse as making vocab lists, writing essays, generating computer programs, producing pop quizzes, and so much more. It is beneficial, quick, and produces exceptionally high-quality findings. ChatGPT replies are quick, free, and often a wonderful place to start for people to create their own [40,41].

Whether ChatGPT belongs in the classroom or not, it is simple to concur that students should be kept securely online. It is crucial in programmes like ChatGPT, where the filters sometimes exclude harmful material. Although content filters on school computers do prevent students from viewing potentially unsafe information, they are simple to get around. We may construct interactive lectures or classes with Chat GPT. For instance, we may pose questions to the chatbot and invite students to react with their responses. It is a fantastic technique to keep students interested in the subject matter. ChatGPT responds based on patterns after being trained on massive quantities of material to comprehend and converse in human language. The ChatGPT can explain grammar-related concepts, teach new language in context, and correct users' errors [42–44].

Another significant capability of ChatGPT is its ability to produce text outlines. Copy the original text into the tool, then briefly explain the desired result. The chatGPT will create a paragraph with excellent organisation and the most important details. Using a vast quantity of data gathered from the internet, ChatGPT uses the third generation of the GPT model to produce text that resembles human answers.

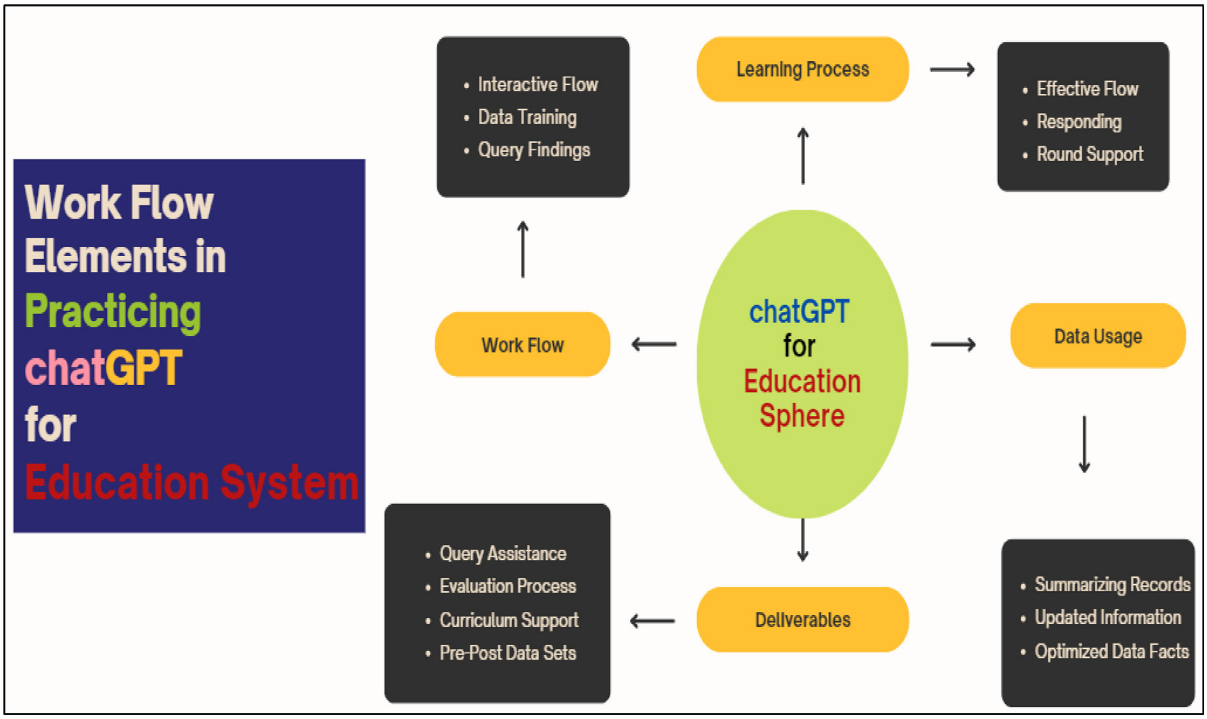


Fig. 2. Typical elements of ChatGPT framework for education domain.

Human feedback helps the bot create better replies that align with human accuracy and natural language standards, thus optimising the system. More than traditional assignments like essays will be required to demonstrate a student's writing prowess. ChatGPT already completes programming homework and produces excellent historical and philosophical writings. Since AI can already do present tasks, evaluating students must drastically alter [45,46].

ChatGPT uses natural language processing (NLP) and deep learning technologies to understand user input in natural human language and produce text exchanges that are human-to-human conversations. Reinforcement Learning from Human Feedback (RLHF) is a technique that has been used to train a big language model to converse and respond to questions as if users were speaking to a natural person. As a result, the computer may analyse and modify its replies in response to input from actual individuals [47,48]. ChatGPT can comprehend the context and meaning of those words and provide the correct answers based on that knowledge. Building chatbots that can have exciting and lifelike discussions with users is achievable using ChatGPT. By training it on data related to that area, ChatGPT may be tailored for specific domains or jobs, such as customer service. As a result, the chatbot may provide replies to user input more precisely and relevantly [49,50].

6. Work flow elements of ChatGPT for education system

Fig. 2 depicts distinguished elements related to ChatGPT structure towards the solicitations in the education domain. To process the chatGPT working structure, a streamlined flow of information and knowledge is a must. It further reflects on several related criteria, services and learning processes, database traits, workflow progress stages, etc. Fig. 2 exemplifies the different working and progressive steps of the chatGPT system for supporting the routine needs of the social structure [51,52].

Instructors may utilise ChatGPT to develop questions for discussion with their students based on a book, subject, historical event, idea, etc. This would allow instructors to swiftly come up with interesting questions for discussions on various aspects of the topic. This may be particularly useful if each student could gain from a different speed. For students to better understand a subject or idea, ChatGPT may provide a variety of examples as well as extra practice opportunities. ChatGPT can modify text for various age groups, so the instructor might either rewrite and provide other examples or ask ChatGPT to clarify it for a younger audience. With the aid of ChatGPT, students may increase their knowledge of new terms or add them to their vocabulary. In this way, it is also a helpful tool for learning new terms. Many ideas are explained in ChatGPT, which might benefit students by giving them in-depth explanations. This can be particularly useful for homework assignments or other circumstances when an instructor is not readily accessible to address quarries [53,54].

ChatGPT is a far superior alternative to any search engine because it can translate documents, regenerate incomplete answers, solve mathematical problems, clarify those concepts, and generate more similar content for practice. Many applications of ChatGPT have been developed, including automated customer service, intelligent virtual assistants, narrative creation for video games and movies, picture captioning systems, summarisation algorithms, and others. The language model ChatGPT is impressive and can completely change how humans interact with computers [55–57]. Its ability to understand and generate text could have significant implications for businesses. ChatGPT uses the most recent developments in natural language processing to process spoken and written language and provide the correct replies. It helps people to communicate with AI systems more effectively and make more informed decisions.

Many instructors now consider using AI-based tools such as ChatGPT rather than trying to avoid it. Students may do this by entering a question into ChatGPT, then reviewing the language the bot generates and evaluating its merits and demerits. With little human input, ChatGPT can compose everything from a high school essay to complex computer programs. Education leaders may utilise ChatGPT as a very effective tool to produce website content. It may speed up the creative process and boost consistency. This provides clear and explicit directions, evaluates and adjusts the output, and utilises ChatGPT as a tool to get the most out of it. With Chat GPT, support and customer service may be provided very cheaply. Businesses may use Chat GPT to reduce the number of customer support agents required to address client inquiries, which leads to cutting down on overhead expenses [58,59].

ChatGPT may provide much potential for development and boost a company's general effectiveness when used correctly. Developing interactive tutoring programmes that can reply to a student's inquiries and provide real-time feedback and direction is another possible use of generative AI-based ChatGPT in education. Ultimately, generative AI has the potential to improve learning by making it more individualised, interactive, and effective [60–62].

ChatGPT is an AI-based tool that uses ML and natural language processing to interact with users. With its intelligent and realistic-sounding interactions, ChatGPT shines. It can understand everyday language and provide accurate responses to queries. Based on each student's unique requirements, interests, and learning preferences, teachers may employ this technology to provide individualised learning experiences for them. This might include using AI to create unique resources or exercises and providing students with real-time feedback and assistance while working [63–65]. Students may collaborate on projects, exchange ideas, and learn from one another when teachers employ AI technology to ease communication and cooperation.

ChatGPT is a cutting-edge language generation model that has the potential to alter how businesses communicate with their customers. Many tasks are used in educational contexts where the ChatGPT is used. Dedicated instructors are doing webinars and producing materials [66,67]. While ChatGPT may be a helpful tool for online learners, it should not be depended upon as the only source of knowledge or assistance. To achieve a well-rounded education, online learners should actively seek various sources and utilise ChatGPT to complement their other learning materials. Several individuals are already considering using ChatGPT to improve education rather than using it for risk management. Many instructors already use it as a teaching tool [68–70].

7. ChatGPT applications in education

ChatGPT can be used as a tool to help students with their studies by creating relevant content and sources on a specific subject. It may also be used to provide students feedback that helps them to improve their knowledge. This might help ensure that students are given the right amount of challenge and material they find interesting and relevant. The ChatGPT model has the potential to be used as a tool for developing summaries, flashcards, or quizzes based on a particular topic or subject area [71–73]. By providing individualised, flexible, and exciting learning opportunities, ChatGPT has the potential to enhance student's educational experiences. The education sector has welcomed this technology as a game-changer. Personalised learning experiences may be supported, as can knowledge gained via research and automation of testing. ChatGPT may be used to create chatbots and virtual assistants that can respond to client inquiries in a conversational manner [74–78]. Further, the significant applications of ChatGPT for education are discussed in Table 1.

ChatGPT may generate text for various uses, such as chatbots and virtual assistants, content production, customer support, language translation, and automated decision-making. ChatGPT utilises many text data to "train" itself, and it then uses that training data to create new text depending on the input. An extensive volume of material from books, papers, and the internet was used to deep-train the model. It uses training data to produce new language that is cohesive, pertinent, and context-aware when given a beginning text as a cue. ChatGPT may be used as a writing assistant to help produce fresh material that is cohesive, pertinent, and aligned with the context. By using plagiarism detection software, educating students, offering resources, and enforcing stringent norms and measures for usage, colleges and institutions are actively combating ChatGPT [79,80]. ChatGPT is intended to look and sound like human interaction. The chatbot can converse on any subject and can even come up with answers to queries. Applications for creative writing might take advantage of ChatGPT's text-generation capabilities. It might come up with writing prompts, offer comments on rough drafts, or even come up with fresh material. This could change how creative writing is taught and practiced [81,82].

ChatGPT has the potential to be a valuable educational tool. It might be used to create course materials, provide task comments, and respond to student inquiries. ChatGPT has a strong command of the English language and can pick up new knowledge, making it a valuable tool for teachers and students. ChatGPT is a tool for rapidly constructing an outline for an article. With its AI-driven natural language processing, it can learn the structure of any post and build an ordered and thorough outline in only a few clicks. The performance and ethical performance of GPT are analysed by DIKWP [83,84]. Language translation may undergo a revolution through the ChatGPT's capacity to comprehend and produce text in various languages. Without human translators, it might enable communication between people and organisations that speak different languages. ChatGPT is an effective technology that may support instructors in personalising instruction, enhancing language proficiency, and facilitating research and writing [85–88]. Educators must keep up with the most recent advancements and consider how they may be utilised to enhance their students' learning as AI develops and becomes more common in the classroom [89,90].

8. Discussion

ChatGPT uses a massive amount of data that it gathers, analyses, and transforms into written sentences. It can write regardless of its kind, structure, or subject. For a variety of disciplines, ChatGPT may provide students with study tools like study guides and flashcards to help them remember key concepts and facts. It may also be used to create examinations and quizzes to ensure students fully comprehend the subject matter. It may also include translations, resources for learning new languages, and tools for enhancing grammar and vocabulary. It may also provide sample test questions and answers, study resources, and flashcards to help students prepare for exams. Incorporating AI into education can improve learning outcomes, make learning more dynamic and exciting and provide students with new learning and development possibilities.

ChatGPT is ideal for its portability, human-like responses, flexibility, and versatility. Due to these features, it is a valuable tool for anybody interested in problems requiring natural language processing. It may be a massive assistance to students doing research, helping them with their assignments, and giving them comments on their work, and it can raise students' knowledge levels across the board. With natural language processing, the bot can understand input from human voices or write without needing menus or programming. By responding to questions about specific subjects, giving prompt and accurate answers, providing additional explanations and clarifications, and developing customised learning plans based on a student's learning preferences, strengths, and weaknesses, this technology can help students in their academic pursuits. It can provide every student with tailored help via individualised learning and interaction.

Students of all ages can easily use ChatGPT as a writing assistant. It can help active learners throughout the writing process by offering suggestions for writing topics, flow ideas, sentence structures, and vocabulary. The interface of ChatGPT makes it possible for students to access thorough and accurate information with their search results. In contrast to most search engines like Google, which give a massive quantity of information with limitless results, ChatGPT provides clear and crisp answers that immediately address the relevant inquiry. Instructors may instruct the programme to generate a variety of phrases, including a new term that the students are unfamiliar with, and then instruct the students to infer the word's meaning from the context of the various sentences.

The programme may produce interesting writing assignments for learners based on age and grade. Instructors might ask ChatGPT to provide a writing exercise or story starter that encourages students to express their creativity to complete the assignment. Students may enhance their reading and comprehension abilities by using ChatGPT. Instructors may instruct the programme to produce passages on various

Table 1
Significant applications of ChatGPT in education.

S No	Applications	Description
1.	Enhance critical thinking and communication abilities	• ChatGPT has the potential to become a crucial tool for writers who wish to improve both their critical thinking and communication abilities. • Students can also use ChatGPT for class assignments and even utilise the bot to create an initial plan. • Subsequently, students may discover how to improve their writing by going beyond the initial draft. • With a wealth of literature supporting its responses, ChatGPT successfully addresses common, basic inquiries on general knowledge, historical events, scientific principles, coding, and fundamental languages. • ChatGPT has exposed many of us to a vast array of opportunities. • In light of this knowledge, it is clear that technology can significantly help students in higher education. • With careful design and implementation, AI can improve student learning outcomes and learning experiences.
2.	Provide instructional material	• ChatGPT will help colleges and universities to provide instructional material. • This system will be able to create customised projects for each student while considering their preferred learning style and current skill level. • Although some students may learn better by visually employing examples, others may need definitions in written form. • This technology may direct students to the proper online materials, such as an e-book, course modules, and assignments, to assist them in improving their understanding of a particular subject. • Depending on their knowledge, they may suggest extra tutoring or advanced preparatory programmes to the student's instructor or school. • ChatGPT quickly gained popularity and was among the top online searches once it was introduced. • ChatGPT can create sentences that are many paragraphs lengthy, correct, comprehensive, and highly exact, as well as tailored to the user's request. • The quality of its responses and the speed at which it interacts with the user are astonishing. • Moreover, it accomplishes it quickly and in several languages.
3.	Conversations with students	• ChatGPT has the potential to start conversations with students in a virtual learning environment. • It may aid in identifying areas of weakness in knowledge and comprehension and assist by recommending workarounds, responding to inquiries, and assisting with content- and context-based search, text creation, and completion to help students get back on track. • It is capable of carrying out operations that typically require human intellect. These operations need language processing, pattern recognition, learning, and decision-making. • As a chatbot or talking computer program, ChatGPT comprehensively creates text. • Every question we have when we visit the site may be asked, whatever comes to mind, and an answer will be given immediately. • The possibilities are endless; it might be information, ideas, or even current affairs. • It is a trained model that may be customised for specific educational jobs. • It has several applications, and the degree of flexibility and precision of its responses is astounding.
4.	Enhance reading and abilities	• Educators may use ChatGPT to develop assignments, question papers, and other learning materials. • Students may enhance their reading and comprehension abilities by using ChatGPT. • Instructors may instruct the programme to produce passages on various subjects, combine its output in classroom assessments, and build questions for students. • This will allow teachers to evaluate how well their students comprehend the subject or topic and pinpoint any areas that need more study. • ChatGPT is a conversational bot that responds to user questions in a way that enables it to search massive databases and to produce well-structured essays, legal briefs, poetry, computer code, or Rogers and Hammerstein song lyrics. • ChatGPT is the greatest AI chatbot ever made available to the general public. • ChatGPT is primarily being met with awe and apprehension, much like the telephone.
5.	Virtual teaching assistants	• ChatGPT may also be trained to serve as virtual teaching assistants to lighten the workload on teachers. • It may be programmed to carry out various educational tasks, including providing onboarding services, helping students, acting as a tutor or mentor, giving feedback, and grading students. • Now, educators, public intellectuals, and academics are having a passionate debate regarding ChatGPT's implications. • There is growing agreement that academics and educators might fall for tricks. The usage of this new technology is already familiar to students. • ChatGPT has a remarkable ability to structure queries and obtain reliable responses. • Observing how quickly teachers adapt to this brand-new classroom situation and appreciate deeper, more engaged learning is encouraging.
6.	Allows students to ask better questions	• ChatGPT may help parents and kids by allowing them to ask questions, start the enrolling process, and encourage further action. • ChatGPT may be further taught to respond to common questions from students and point them in the direction of appropriate resources. • This technology may gather student comments and other valuable data, which instructors can evaluate and utilise to enhance their teaching and learning strategies and development goals. • ChatGPT is a powerful AI technology that enables voice or full-sentence web searches. • Instead of the usual Google Search results, the searcher is presented with the results in detailed, in-depth phrases by using this technology. • The tool has gained popularity in education since it can write essays, provide in-depth answers to queries, and close learning gaps by utilising digital resources and AI. • ChatGPT is a potent language model that may be utilised to produce text for chatbot applications that sounds like human speech. • Businesses may enhance customer service, simplify processes, and provide clients with individualised advice using ChatGPT's features.
7.	Understands complex problems	• ChatGPT understands complex problems better than other contemporary, accessible AI systems, making it the popular choice for handling complex queries. • Technologies like ChatGPT may aid in creating chatbots and virtual assistants for use in education. With its ability to reimagine teaching and learning, • ChatGPT offers a chance to influence the future of the classroom. • Conversational AI is expected to alter how parents, instructors, and students communicate. • By handling routine tasks, an AI tool similar to ChatGPT may significantly improve the learning experience on our digital learning platform. • From facilitating the onboarding of new students or teachers to offering individualised, self-paced instruction depending on the learning preferences of each student. • In addition to these tasks, this may help resolve frequently asked questions, gather information and feedback to enhance teaching methods, and monitor class and student performance.

(*continued on next page*)

Table 1 (*continued*).

S No	Applications	Description
8.	Straightforward response	• ChatGPT can respond to queries straightforwardly; it can write code, make lists, react to emails for us, and even answer our queries. • It can produce detailed and human-like text, interpret human speech, correct grammatical errors, and question false premises. • The programme may produce interesting writing assignments for students based on age and grade. • For instance, instructors might ask ChatGPT to develop a writing exercise or story starter that encourages students to express their creativity to complete the job. • It may be a first step in teaching students how to write. • By introducing new words and helping them become the foundation of sentences, ChatGPT may aid students in growing their vocabulary. • Instructors may instruct the programme to generate a variety of phrases, including a new term that the students are unfamiliar with, and then instruct the students to infer the word's meaning from the context of the various sentences.
9.	Topic brainstorming and creativity	• It can assist students with grammatical correction, topic brainstorming, and creativity when developing project ideas. • Writing lesson plans, emails, or even letters of recommendation for other teachers may assist instructors in lightening their burden. • ChatGPT may increase instructor productivity and facilitate student learning. • Although a ChatGPT cannot replace a teacher, it may allow instructors to interact more with students. • By offering ideas and questions for students to reflect on, ChatGPT may aid in facilitating dialogues and fostering critical thinking. • For instance, we may list open-ended questions on a specific subject using ChatGPT and then ask students to debate and react to these questions in small groups or as a class. • Students' critical thinking abilities and comprehension of the subject matter may benefit from this. • ChatGPT can automate repetitive tasks like delivering product details and question responses.
10.	Enhance learning personalisation	• ChatGPT will enhance learning personalisation and ultimately become an essential component of the learning process. • We must give students the tools they need to harness this power to better prepare them for the future. • ChatGPT information may be used by businesses to improve their offerings and meet client requirements. • Because of its natural language processing capabilities, which enable it to ascertain what clients believe about a product, ChatGPT may generate leads by talking with prospective customers and learning about their requirements. • Based on the interests and preferences of each consumer, we may utilise this information to tailor our marketing efforts. • Depending on the student's preferences, ChatGPT may provide tailored suggestions for each.
11.	Text analysis	• ChatGPT can often provide us with some entirely accurate replies to our inquiries. • It is both exciting and terrifying. • In essence, it is a learning engine that has been "trained" to spot patterns in text collected from websites worldwide and combined with AI to produce responses that seem authentically human. • Language translation is one of ChatGPT's most potential uses. • The model is an effective machine translation tool since it can comprehend and produce text in various languages. • The algorithm can learn to accurately translate text from one language to another by being fine-tuned on a large dataset of bilingual material. • This may be used for various purposes, including translating chatbots, websites, and documents. • Text summary, which extracts the most crucial details from a lengthy text, is a function of ChatGPT. • This may be helpful for several uses, including summarising news, product reviews, and research papers. • It may also be used for text analysis tasks, including named entity identification, topic modelling, and sentiment analysis.
12.	Craft essays	• ChatGPT can develop essays, poems, questions, answers, and computer code. • AI text systems may rapidly generate text, and it is often difficult to tell them apart from human-written text. • ChatGPT can produce writing that resembles that of a person. • This model is capable of responding to a prompt with a comprehensive answer. • The model can comprehend and reply to various subjects and inquiries since it has been trained on vast text data. • The field of education, especially in college-level learning, is one of ChatGPT's most important effects. • With the increased technology usage in the classroom, ChatGPT may be a potent tool for improving students' learning experiences. • As a learning aid, ChatGPT can be utilised in higher education. • Students may get prompt and precise information by creating query replies using the model. • This might be very helpful for students with trouble grasping a particular idea or subject.
13.	Enhances the learning environment	• This enhances the learning environment in the classroom. • Using the bot to administer tests is the best method a teacher may utilise ChatGPT in the classroom. • Thus, to test students' understanding, the AI chatbot may provide straightforward yes/no or more difficult multiple-choice questions on a subject. • Teachers may devote more time to lesson preparation and student engagement by utilising ChatGPT to construct exams and quizzes. • As a writing assistance, ChatGPT may also be used in higher education. • Those who struggle with writing or need to generate a lot of written work quickly may find this extremely beneficial. • ChatGPT may be utilised in various businesses, including journalism, customer service, and more, in addition to education. • The methodology may be used to produce software code, news articles, and even customer support routines. • This makes it an essential tool for enterprises since it enhances productivity and speeds up the completion of activities.
14.	Understand and communicate languages	• With the help of ChatGPT, students may easily understand and communicate in other languages. • Moreover, it may provide resources like dictionaries and grammatical rules for learning other languages. • ChatGPT can respond to most computer science questions and tasks studied in school. • The teachers can suggest a variety of ChatGPT-based tasks that can be assigned to computer science students to emphasise that computer science thinking skills have not become obsolete. • It may improve students' computer science thinking abilities and expand their comprehension of computer science topics. • ChatGPT is one of the most effective chatbots because it can learn in real-time and recall user indications from past talks. • This technology already has a wide range of uses beyond simple question–answering. For instance, ChatGPT has been required to provide scholarly papers, code, and emails.

(*continued on next page*)

subjects, combine its output in classroom assessments, and build questions for students. This will allow teachers to evaluate how well their students comprehend the subject or topic and pinpoint any areas that need more study.

ChatGPT can respond to follow-up inquiries, acknowledge errors, refute unfounded assumptions, and reject improper requests throughout the conversation. This technology can generate text and translate it across languages. Its primary characteristic is the capacity to generate

Table 1 (*continued*).

S No	Applications	Description
15.	Boost exam preparation	• ChatGPT may be a helpful resource for students to assist with homework and other tasks, practice language skills, and boost exam preparation. • It may help students save time and effort by quickly summarising books and articles, providing arguments and examples, and aiding in research and writing. • By having it produce arithmetic problems or questions for students to work on together, we may utilise ChatGPT to support group collaboration. • This is an excellent technique to promote teamwork and problem-solving abilities. • This may be an entertaining and exciting approach to studying the content while fostering competitiveness. • ChatGPT could respond to questions with remarkable fluency and coherence using AI, and among other things, it might pass muster as a well-written answer to a class assignment. • ChatGPT might increase the time spent writing in class as the instructor coaches and consults rather than merely discouraging or monitoring AI help. • ChatGPT provides a mechanism to broaden the focus and complexity of its courses.
16.	Exact information	• Students may get exact information and receive results right away using ChatGPT. • Students may need assistance narrowing the scope of the information they initially needed due to the abundance of Google results. • The replies given in the instance of ChatGPT are logical and comprehensive. • For instance, ChatGPT may assist a student with maths problems by solving the issue, illustrating the underlying ideas, and producing other issues based on the same idea for practice. • Critical thinking instruction might be enhanced with the use of ChatGPT. • Today's teachers include listening, talking, and engaging in constructive arguments while teaching writing and English. • Writing, however, only assists sure students in organising the knowledge they acquire. • Instructors could collaborate with Chat GPT to enhance kids' cognitive abilities. • Students should eventually be able to use AI technologies to learn new facts.
17.	Save instructor time	• To save time and energy, instructors may ask ChatGPT to produce assignments that fill the knowledge gaps before introducing specific ideas or to develop longer lessons for better understanding. • ChatGPT is merely developed to produce words in response to input. • It can spout lengthy responses, indicating that the depth and insight in its responses are likely to be lacking. • Technology can be used for good deeds and constructive social change. • ChatGPT and related language models will become more common and powerful. • They should be viewed as tools that supplement and improve human expertise rather than as a replacement for it. • With ChatGPT's ability to type almost anything, it is debatable if students still need to learn how to write. • Many may be curious whether AI technology will ever entirely replace writing. • With ChatGPT's assistance, educators can emphasise creative idea organisation, revision, debate, and critical thinking.
18.	Research tool	• ChatGPT may be used as a research tool to come up with answers to questions or prompts on a particular subject. • Use ChatGPT, for instance, to come up with answers to open-ended questions or prompts on a particular subject of study, such as psychology, history, etc. • This could help develop ideas or research many viewpoints on a particular subject. • The enormous potential of ChatGPT may be assessed by its capacity to reply to particular messages and adjust to ongoing conversations. • The messages become improved to a more significant extent over time as the system continues to engage with the user. • Also, it has enormous potential to provide improved customer service by efficiently responding to client inquiries. • ChatGPT and related technologies are potent language models that have the potential to change how humans communicate with computers entirely.
19.	Summarise large documents	• ChatGPT may be used to summarise large documents or articles. • This may be used to quickly get a broad idea of a book without reading it. • ChatGPT may be used to assess the emotional content of a text. • It can be used to evaluate the tone of customer reviews and ascertain the overall mood and emotion of a piece of writing to raise customer satisfaction. • ChatGPT is a conversational language model that produces text that resembles human speech depending on input using deep learning algorithms. • It has been trained on various online content and can provide high-quality, coherent responses to various inquiries and prompts. • With access to millions of online repositories and resources, ChatGPT can provide a solution to the quarries. • A substantial amount of text data was used to create the ChatGPT language model. • The model may produce remarkably accurate and fluent responses in response to various natural language processing tasks.
20.	Evaluation of student performance	• Other areas where ChatGPT may have a significant impact include assessing and evaluating student performance. • Once trained, the AI chatbot can grade lengthy essays according to predetermined standards like content, style, and organisation. • Further, it also provides students feedback to help them become better writers. • Due to its ability to interpret natural language and provide meaningful replies, ChatGPT could be used to create more effective assessment and evaluation methods. • ChatGPT provides perceptive thoughts on current events or other fascinating subjects. • ChatGPT can swiftly resolve any level of mathematical problems we provide it because of its AI algorithms and mathematical expertise. • We may ask the chatbot to do integral or derivative calculations, simplify algebraic phrases, or compute complex formulae. • It is also helpful for instructors who need additional tools to successfully and efficiently teach mathematics to their students. • ChatGPT uses AI to create conversations that resemble those between people automatically. • ChatGPT has a variety of functions, including intent detection, emotion identification, answer customisation, and others.

(*continued on next page*)

text that resembles human writing in response to predetermined cues. Students learning a new language might benefit from conversation practice and feedback via ChatGPT. In order to encourage students to engage in conversations and cooperate, ChatGPT or other models can offer instructions for group projects and assignments. ChatGPT is a newly developed language model that can provide human-like replies to various queries and prompts after being trained on a massive quantity of text data from the internet. As a result, ChatGPT may be used for various purposes, such as chatbots, text production, and language translation.

As one of ChatGPT's primary characteristics, it can produce text based on patterns identified in the data it has been trained on. This

Table 1 (*continued*).

S No	Applications	Description
21.	Automatic grading systems	• ChatGPT may be used to create automatic grading systems, reducing the workload on instructors and providing students with faster, more precise feedback on their performance. • As a result of ChatGPT's real-time comprehension and response capabilities, interactive tests may be created where feedback can be tailored to the needs of students. • This makes learning more engaging and interactive and identifies areas where students want further support and guidance. • ChatGPT can grade assignments more accurately than a busy instructor who is only sometimes compensated for doing it, which is good. Instead of just marking assignments, teachers should concentrate on engaging and inspiring their students. • When given pertinent terms as first instructions, it can generate fresh ideas, which might help us unleash our creative talent. • Because of the massive data, ChatGPT can help you think creatively and outside the box. • ChatGPT is an ML model that can generate reactions to various signals that resemble those of humans. ChatGPT excels in natural language processing tasks, such as text summarisation, response generation, and language translation.
22.	E-learning	• In the world of e-learning, ChatGPT has fundamentally transformed the game's rules. • It can give students fast and accurate information, increasing the effectiveness and efficiency of e-learning platforms and virtual learning courses. • This implies that Ed-tech businesses are adopting ChatGPT to support students as they progress through e-learning courses and to offer more details and explanations when there are few opportunities for student-teacher interactions. • ChatGPT can compile all the knowledge needed to resolve the issue, saving the student from doing additional research. • In-depth explanations and examples for various concepts and topics may be provided through ChatGPT, assisting students in understanding challenging material. • Students needing more support or having difficulty with a specific topic may find this extremely beneficial. • ChatGPT can help students with their research by providing relevant information and resources. • Also, it may help students edit and revise their written work, which helps them progressively become better at writing.
23.	Interactive experience	• ChatGPT's natural language understanding might create a more engaging and interactive e-learning experience. • ChatGPT might provide virtual instructors and study tools, fostering a more individualised and exciting learning environment. • Using ChatGPT, instructors may experiment with innovative approaches to streamline their job and lighten their burden through advancements in AI and ML. • It enables students to share knowledge organically in a social media setting. • ChatGPT may help students improve their grammar and vocabulary by describing grammatical concepts and providing practice challenges. • It also helps students find and correct common grammar mistakes in their written work. • Students may prepare for tests and hone their test-taking skills using ChatGPT to generate practice test questions and answers for different courses and exams. • ChatGPT can produce highly relevant responses to the question and exhibit a degree of understanding and knowledge comparable to a human's. • This makes the paradigm particularly useful for activities like authoring documents and translating across languages.
24.	Online education	• ChatGPT is particularly beneficial in the context of online education. ChatGPT is a tool that can provide individualised, interactive learning experiences in the age of growing online learning. • For instance, students may utilise ChatGPT to ask questions and instantly get answers regarding subjects they are learning. • It may lessen the need for conventional teaching techniques and enable students to progress at their learning rate. • ChatGPT can tackle the minor issues that stop many students' learning in their tracks. • The tool may assist kids with their tasks, homework, and learning challenges. • For instance, because the tool can make essays, professors might instruct students to use it to create many essays on the same topic or subject and compare the ideas provided by the AI. • This would aid kids in developing critical and creative thinking abilities and working on comprehension, reading, and writing abilities. • ChatGPT can provide students with individual feedback and assistance, responding to inquiries and explaining various academic subjects.
25.	Assist in preparing debates	• It assists students in preparing for debates by coming up with arguments and refutations on a particular subject. • This creates ideas, outlines, and even finished speeches to assist students in preparing talks. • Using ChatGPT to write essays also has the advantage of producing grammatically sound and coherent sentences. • Students who need help with grammar and sentence structure may find this helpful. • The learner may use ChatGPT as a starting point and then edit and rewrite the content produced to fit their writing voice and style. • ChatGPT is an AI language model having access to a massive quantity of data that was trained on millions of pages of data. • By applying natural language processing algorithms, it may utilise this data to generate answers to questions and instructions people enter. • ChatGPT employs a wide range of possible applications, including aiding in the teaching and learning of languages. • ChatGPT often offers short responses quickly and roughly precisely, but a Google search might be frustrating due to the multiplicity of diverse voices.
26.	Enhance knowledge	• This might include assisting students in enhancing knowledge and abilities in meaningful ways by employing simulations, virtual worlds, or other interactive tools. • Instructors may design engaging, practical problem-solving tasks for students by using this technology. • Instructors may utilise their time with students to provide those who need the most customised feedback and help. • This might include giving students individualised support and direction while working with them one-on-one or in small groups. • ChatGPT could overlook inevitable mistakes when presented with a text that is full of them. • ChatGPT may be used in the classroom to provide students with individualised learning experiences to keep them engaged and inspired to finish their studies. • ChatGPT was created mainly for conversational and chat-based applications, and it can comprehend user inputs and produce text that resembles human speech. • This makes it helpful for applications like chatbots, virtual assistants, and other conversational AI systems.

(*continued on next page*)

implies that although it may provide responses that resemble language found in the training data, it might not necessarily produce ethical or correct responses. Students should be aware of the limits of AI chatbots and critically assess the results produced. The usefulness of various chatbot interfaces or design elements may also be researched using ChatGPT. It may be beneficial for researchers to examine how users engage with ChatGPT and compare the findings to those of other chatbots to gain additional insight into how to create an efficient and

Table 1 (*continued*).

S No	Applications	Description
27.	Advice to students for better interview	• ChatGPT offers advice to students on how to be ready for an interview. • After practising using ChatGPT, they get feedback and suggestions for progress. Students question ChatGPT about life. • A student asks ChatGPT for advice and ideas on improving their communication skills. • ChatGPT can review their code for flaws and provide feedback for improvements for students learning how to develop a website. • A teacher must see ChatGPT as an additional tool to the other tools and abilities acquired through education and practice. • The ChatGPT model can be utilised as a tool for text translation across languages. • Students learning a new language might use the model to translate things written in another language into their mother tongue since the tool can recognise and create numerous languages. • The provision of accurate and accessible translation tools, which can aid students in better comprehending and engaging with foreign-language materials, has the potential to enhance the language learning experience.

exciting chatbot experience. Using ChatGPT as a source of knowledge may make students less likely to engage in independent learning and critical thought, making them more reliant on AI for solutions.

Although there is some debate on the usefulness of an AI chatbot, ChatGPT is undoubtedly growing in popularity since it offers more conversational responses than humans. The ChatGPT may also be used to learn about various subjects, summarise lengthy articles and papers, translate different languages, create tales and poetry, help with coding, and more. Creative tales and other text-based material may also be produced with it. ChatGPT is an excellent option for companies seeking to improve customer service and for developers who want to have more engaging interactions with their customers. ChatGPT helps create customer service chatbots, providing replies to questions in online forums and even developing personalised content for social media postings since it can create text responses that mimic those of people in response to instructions. The ChatGPT model can translate the information organically and adequately when given a text prompt in the source and target languages. ChatGPT can execute various jobs, including composing emails and tales, essays, making music, conversing with people, paraphrasing, computer coding and decoding, software programming, and much more, despite being pre-programmed to simulate human communication. It helps summarise, and depending on our demands, it could help us condense our projects and research into a certain number of words. This enables the model to perform several natural language tasks with high accuracy and fluency, including text creation and translation.

9. Limitations of ChatGPT in education

The evident face of ChatGPT, its ability to respond to questions, raises some concerns about the legitimacy of lessons and homework assignments. One of the most common worries in the education sector is that students will use ChatGPT to finish their homework and then copy and paste the solutions without the teacher having any control. Several colleges and institutions outlawed the use of this technology for writing tasks as students began utilising it to compose their homework, essays, and theses. It is harder and less reliable to find if this AI-generated material is the same or different from the plagiarised text. ChatGPT is a freakishly powerful instrument that works well across various chores and academic disciplines. AI-generated writing raises ethical issues, and there are worries regarding the veracity of ChatGPT's responses. Using ChatGPT and other language models raises crucial ethical issues about its effects on society.

ChatGPT may sometimes create inaccurate information and provide damaging instructions on biased material on its webpage. While creating text, a chatbot automatically adds words that are most likely to come after the previous words; nevertheless, it does not verify the truth of the information. The possible bias of the data the bot is trained on is a crucial ethical problem associated with using ChatGPT. Any biases in the chatbot's training dataset are reflected in the model's output, which might lead to incorrect or dangerous information. Although the opener has various safeguards to prevent users from abusing the conversation, the issue of unfair, sexist, racist, and other objectionable comments still exist. Due to a lack of practical applications and a limited grasp of the technology, institutions need help to define rules and procedures connected to ChatGPT.

The data is constantly being churned from the cloud, so it cannot receive information from a particular source. Its excellence resides in its capacity to synthesise data from several sources and provide mostly original answers to the same query. ChatGPT is similar to any expert system or information-limited mobile application if we depend on it to obtain data from a particular source. While it may compensate for the lack of employees by offering users a complete source of information, it is up to the individual user to make the most of this technology's potential and exercise prudence when using the knowledge. It works with a limited dataset that needs to reflect the present accurately. This raises the likelihood of producing inaccurate information. Concerns about privacy, data security, and intellectual property are among the moral and legal issues that the use of AI in education brings up. To guarantee adherence to laws and moral norms, it will be necessary to carefully evaluate these concerns before using ChatGPT in higher education.

ChatGPT users should be conscious of the potential for bias in their replies and take steps to minimise it. Some people are concerned that ChatGPT will eliminate jobs for writers, marketers, and other professionals who rely heavily on written communication, like when machines and computers replace human labour. Developers have used it to overcome coding difficulties. ChatGPT functions as an AI learning model and needs access to such data. Moreover, ChatGPT may give its customers incorrect answers since it is only as good as the data it was trained on. Students' and staff members' privacy may be in danger if the data used to train the model is not sufficiently anonymised or protected. Since ChatGPT communicates with users across a network, data from users may be intercepted, accessed, or altered by attackers. ChatGPT might be incorporated into a social engineering attack to access confidential data. The platform may assist attackers in gaining the user's confidence and gathering data that may be exploited for harmful purposes, such as creating credentials.

ChatGPT is an effective tool that can produce text replies to various queries, including some that can include sensitive or private information. As a result, utilising ChatGPT has several dangers, including the chance that it could produce inappropriate or offencive material, leak private information, or be influenced by nefarious individuals. The ChatGPT's response may differ depending on how the input is worded or how often the same prompt is issued. The model may only sometimes know the answer or may only sometimes provide the correct response. Due to biases in the training data and over-optimisation, the ChatGPT model may be wordy and misuse specific phrases.

10. Future scope

In the future, we can use ChatGPT web search to do in-depth market research online. It may work along with other technologies to create our website. ChatGPT can construct homework more effectively than a busy instructor who repeats publicly available online assignment templates. The future will be less frightening and more fascinating for instructors who comprehend AI and use it to their benefit. AI will have the potential to significantly reduce the amount of time instructors spend grading assignments, customising lesson plans, and filling out reports. Instructors who spend time interacting with, encouraging, and helping students will have a more significant impact and maintain their enthusiasm for the subject.

Chat GPT, a highly sophisticated Al tool that can provide almost limitless and all-encompassing service and information on any topic following our desires, might be considered the future of Al in the world. In the future, any student may ask ChatGPT to write an essay on any subject, and the software will comply. Teachers would find identifying text produced by AI simpler if chatbots were trained to watermark their outputs somehow. ChatGPT is suitable for chatbot and conversational AI applications due to its natural language interpretation and creative ability. It might be better to train it on a conversational text dataset so it learns how to comprehend and respond to user input like a natural person.

11. Conclusion

ChatGPT employs deep learning and natural language processing to produce replies to text-based inputs that resemble a person's. ChatGPT is beneficial in education as it is used for various purposes, including language translation, discussion, summarisation, and text production. It is a technology becoming increasingly well-liked in various disciplines, including research and education, through its capacity to learn from vast volumes of data and provide high-quality results. The AI chatbot will influence tutoring and personalised learning in two critical areas. Since ChatGPT uses natural language processing to interpret and reply to inquiries in real-time, it may be utilised to provide students with on-demand, live tutoring. ChatGPT may serve as a virtual teaching assistant by giving students immediate feedback. Students may also use ChatGPT to ask questions to obtain clarification on certain course subjects or to have everything explained to them multiple times. ChatGPT may assist teachers in creating material, including numerous test versions, student learning evaluations, syllabi, rubrics, and more. This technology can provide a satisfactory answer to a challenge or assignment rapidly; it should be revised to enable students to apply their knowledge and abilities to accomplish the task effectively. GPT models may produce grammatically and structurally sound natural language text since they are trained on vast volumes of text data. ChatGPT has been taught to produce more conversational text for chatbot applications. It may start conversations, respond to user input, and provide users with information and support in a chatbot environment. ChatGPT is getting more intelligent and capable of managing complicated jobs as AI advances.

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. Tlili, B. Shehata, M.A. Adarkwah, A. Bozkurt, D.T. Hickey, R. Huang, B. Agyemang, What if the devil is my guardian angel: ChatGPT is a case study of using chatbots in education, Smart Learn. Environ. 10 (1) (2023) 15.

[2] 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, 2023.

[3] A.B. Mbakwe, I. Lourentzou, L.A. Celi, O.J. Mechanic, A. Dagan, ChatGPT passing USMLE shines a spotlight on the flaws of medical education, PLoS Digit. Health 2 (2) (2023) e0000205.

[4] 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.

[5] E. Kasneci, K. Seßler, S. Küchemann, M. Bannert, D. Dementieva, F. Fischer ..., G. Kasneci, ChatGPT for good? On opportunities and challenges of large language models for education, Learn. Individ. Differ. 103 (2023) 102274.

[6] 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).

[7] T.H. Kung, M. Cheatham, A. Medenilla, C. Sillos, L. De Leon, C. Elepaño ., V. Tseng, Performance of ChatGPT on USMLE: Potential for AI-assisted medical education using large language models, PLoS Digit. Health 2 (2) (2023) e0000198.

[8] X. Zhai, ChatGPT user experience: Implications for education, 2022, Available at SSRN 4312418.

[9] 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.

[10] G. Eysenbach, The role of ChatGPT, generative language models, and artificial intelligence in medical education: a conversation with ChatGPT and a call for papers, JMIR Med. Educ. 9 (1) (2023) e46885.

[11] L. Bishop, A computer wrote this paper: What ChatGpt means for education, research, and writing, Res. Writ. (2023).

[12] J.V. Pavlik, Collaborating with ChatGPT: Considering the implications of generative artificial intelligence for journalism and media education, J. Mass Commun. Educ. (2023) 10776958221149577.

[13] X. Zhai, ChatGPT for next-generation science learning, 2023, Available at SSRN 4331313.

[14] Y.K. Dwivedi, N. Kshetri, L. Hughes, E.L. Slade, A. Jeyaraj, A.K. Kar ., R. Wright, So what if chatgpt wrote it? Multidisciplinary perspectives on opportunities, challenges and implications of generative conversational AI for research, practice and policy, Int. J. Inf. Manage. 71 (2023) 102642.

[15] M.U. Haque, I. Dharmadasa, Z.T. Sworna, R.N. Rajapakse, H. Ahmad, I think this is the most disruptive technology: Exploring sentiments of ChatGPT early adopters using Twitter data, 2022, arXiv preprint arXiv:2212.05856.

[16] M. Halaweh, ChatGPT in education: Strategies for responsible implementation, Contemp. Educ. Technol. 15 (2) (2023).

[17] Y. Bang, S. Cahyawijaya, N. Lee, W. Dai, D. Su, B. Wilie ., P. Fung, A multitask, multilingual, multimodal evaluation of ChatGPT on reasoning, hallucination, and interactivity, 2023, arXiv preprint arXiv:2302.04023.

[18] S. Mitrović, D. Andreoletti, O. Ayoub, Chatgpt or human? Detect and explain. Explaining decisions of a machine learning model for detecting short ChatGPT-generated text, 2023, arXiv preprint arXiv:2301.13852.

[19] F.C. Kitamura, ChatGPT is shaping the future of medical writing but still requires human judgment, Radiology (2023) 230171.

[20] B.D. Lund, T. Wang, Chatting about ChatGPT: How may AI and GPT Impact Academia and Libraries? Library Hi Tech News, 2023.

[21] A. Haleem, M. Javaid, R.P. Singh, An era of ChatGPT as a significant futuristic support tool: A study on features, abilities, and challenges, BenchCouncil Trans. Benchmarks Stand. Eval. (2023) 100089.

[22] F.Y. Wang, Q. Miao, X. Li, X. Wang, Y. Lin, What does chatGPT say: the DAO from algorithmic intelligence to linguistic intelligence? IEEE/CAA J. Autom. Sin. 10 (3) (2023) 575–579.

[23] S. Sok, K. Heng, ChatGPT for education and research: A review of benefits and risks, 2023, Available at SSRN 4378735.

[24] M. Perkins, Academic integrity considerations of AI Large Language Models in the post-pandemic era: ChatGPT and beyond, J. Univ. Teach. Learn. Pract. 20 (2) (2023) 07.

[25] S. Shahriar, K. Hayawi, Let's have a chat! A conversation with ChatGPT: Technology, applications, and limitations, 2023, arXiv preprint arXiv:2302.13817.

[26] A. Lecler, L. Duron, P. Soyer, Revolutionising radiology with GPT-based models: current applications, future possibilities and limitations of ChatGPT, Diagn. Interv. Imaging (2023).

[27] G. Cooper, Examining science education in ChatGPT: An exploratory study of generative artificial intelligence, J. Sci. Educ. Technol. (2023) 1–9.

[28] A. Bozkurt, J. Xiao, S. Lambert, A. Pazurek, H. Crompton, S. Koseoglu ., P. Jandrić, Speculative futures on ChatGPT and generative artificial intelligence (AI): A collective reflection from the educational landscape, Asian J. Distance Educ. (2023) Early-access.

[29] H. Alkaissi, S.I. McFarlane, Artificial hallucinations in ChatGPT: implications in scientific writing, Cureus 15 (2) (2023).

[30] P.A. Rospigliosi, Artificial intelligence in teaching and learning: What questions should we ask of ChatGPT? Interact. Learn. Environ. 31 (1) (2023) 1–3.

[31] T.J. Chen, ChatGPT and other artificial intelligence applications speed up scientific writing, J. Chin. Med. Assoc. (2023) 10–1097.

[32] 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.

[33] 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, (2023-03), 2023.

[34] B. Rathore, Future of AI & generation alpha: ChatGPT beyond boundaries, Eduzone: Int. Peer Rev./Refer. Multidiscip. J. 12 (1) (2023) 63–68.

[35] J. Kocoń, I. Cichecki, O. Kaszyca, M. Kochanek, D. Szydło, J. Baran ., P. Kazienko, ChatGPT: Jack of all trades, master of none, 2023, arXiv preprint arXiv:2302.10724.

[36] 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.

[37] J. Homolak, Opportunities and risks of ChatGPT in medicine, science, and academic publishing: a modern promethean dilemma, Croatian Med. J. 64 (1) (2023) 1–3.

[38] S. Badini, S. Regondi, E. Frontoni, R. Pugliese, Assessing the capabilities of ChatGPT to improve additive manufacturing troubleshooting, Adv. Ind. Eng. Polym. Res. (2023).

[39] M. Koo, The importance of proper use of ChatGPT in medical writing, Radiology (2023) 230312.

[40] C. Zielinski, M. Winker, R. Aggarwal, L. Ferris, M. Heinemann, J.F. Lapeña ., L. Citrome, Chatbots, ChatGPT, and scholarly manuscripts-WAME recommendations on ChatGPT and chatbots in relation to scholarly publications, Afro-Egypt. J. Infect. Endemic Dis. 13 (1) (2023) 75–79.

[41] B. Williamson, F. Macgilchrist, J. Potter, Re-examining AI, automation and datafication in education, Learn. Media Technol. 48 (1) (2023) 1–5.

[42] W.C.H. Hong, The impact of ChatGPT on foreign language teaching and learning: opportunities in education and research, J. Educ. Technol. Innov. 3 (1) (2023).

[43] V.L. Bommineni, S. Bhagwagar, D. Balcarcel, C. Davazitkos, D. Boyer, Performance of ChatGPT on the MCAT: The road to personalised and equitable premedical learning, MedRxiv (2023) 2023-2003.

[44] M. Aljanabi, M. Ghazi, A.H. Ali, S.A. Abed, ChatGpt: Open possibilities, Iraqi J. Comput. Sci. Math. 4 (1) (2023) 62–64.

[45] A. Thurzo, M. Strunga, R. Urban, J. Surovková, K.I. Afrashtehfar, Impact of artificial intelligence on dental education: A review and guide for curriculum update, Educ. Sci. 13 (2) (2023) 150.

[46] M. Sullivan, A. Kelly, P. McLaughlin, ChatGPT in higher education: Considerations for academic integrity and student learning, J. Appl. Learn. Teach. 6 (1) (2023).

[47] T. Teubner, C.M. Flath, C. Weinhardt, W. van der Aalst, O. Hinz, Welcome to the era of ChatGPT others, the prospects of large language models, Bus. Inf. Syst. Eng. (2023) 1–7.

[48] R. Firaina, D. Sulisworo, Exploring the usage of ChatGPT in higher education: Frequency and impact on productivity, Bul. Edukasi Indones. 2 (01) (2023) 67–74.

[49] S. Biswas, ChatGPT and the future of medical writing, Radiology (2023) 223312.

[50] 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.

[51] P. Hacker, A. Engel, M. Mauer, Regulating ChatGPT and other large generative AI models, 2023, arXiv preprint arXiv:2302.02337.

[52] A. Zarifhonarvar, Economics of ChatGPT: A labor market view on the occupational impact of artificial intelligence, 2023, Available at SSRN 4350925.

[53] L.J. Quintans-Júnior, R.Q. Gurgel, A.A.D.S. Araújo, D. Correia, P.R. Martins-Filho, ChatGPT: the new panacea of the academic world, Rev. Soc. Bras. Med. Trop. 56 (2023) e0060–2023.

[54] J. White, Q. Fu, S. Hays, M. Sandborn, C. Olea, H. Gilbert ., D.C. Schmidt, A prompt pattern catalogue to enhance prompt engineering with chatbot, 2023, arXiv preprint arXiv:2302.11382.

[55] A. Ahmad, M. Waseem, P. Liang, M. Fehmideh, M.S. Aktar, T. Mikkonen, Towards human-bot collaborative software architecting with ChatGPT, 2023, arXiv preprint arXiv:2302.14600.

[56] J.K.M. Ali, M.A.A. Shamsan, T.A. Hezam, A.A. Mohammed, Impact of ChatGPT on learning motivation: Teachers and students' voices, J. Engl. Stud. Arabia Felix 2 (1) (2023) 41–49.

[57] M. Giunti, F.G. Garavaglia, R. Giuntini, S. Pinna, G. Sergioli, Chatgpt prospective student at medical school, 2023, Available at SSRN 4378743.

[58] D. Singh, ChatGPT: A new approach to revolutionise organisations, Int. J. New Media Stud. (IJNMS) 10 (1) (2023) 57–63.

[59] P. Fernandez, Through the Looking Glass: Envisioning New Library Technologies AI-Text Generators as Explained by ChatGPT, Library Hi Tech News, 2023.

[60] U. Bukar, M.S. Sayeed, S.F.A. Razak, S. Yogarayan, O.A. Amodu, Text analysis of chatGPT as a tool for academic progress or exploitation. Available at SSRN 4381394.

[61] E. Bonsu, D. Baffour-Koduah, From the consumers' side: Determining students' perception and intention to use ChatGPTin ghanaian higher education, 2023, Available at SSRN 4387107.

[62] T. Sakirin, R.B. Said, User preferences for ChatGPT-powered conversational interfaces versus traditional methods, Mesopotamian J. Comput. Sci. 2023 (2023) 24–31.

[63] Y. Tan, D. Min, Y. Li, W. Li, N. Hu, Y. Chen, G. Qi, Evaluation of ChatGPT as a question answering system for answering complex questions, 2023, arXiv preprint arXiv:2303.07992.

[64] A.M. Hopkins, J.M. Logan, G. Kichenadasse, M.J. Sorich, Artificial intelligence chatbots will revolutionise how cancer patients access information: ChatGPT represents a paradigm shift, JNCI Cancer Spectr. 7 (2) (2023) pkad010.

[65] B. Rathore, Future of textile: Sustainable manufacturing & prediction via ChatGPT, Eduzone: Int. Peer Rev./Refer. Multidiscip. J. 12 (1) (2023) 52–62.

[66] H. Dai, Z. Liu, W. Liao, X. Huang, Z. Wu, L. Zhao ., X. Li, ChatAug: Leveraging ChatGPT for text data augmentation, 2023, arXiv preprint arXiv:2302.13007.

[67] M.A. AlAfnan, S. Dishari, M. Jovic, K. Lomidze, ChatGPT as an educational tool: Opportunities, challenges, and recommendations for communication, business writing, and composition courses, J. Artif. Intell. Technol. (2023).

[68] M. Aljanabi, ChatGPT: Future directions and open possibilities, Mesopotamian J. CyberSecur. 2023 (2023) 16–17.

[69] E. Opara, A. Mfon-Ette Theresa, T.C. Aduke, ChatGPT for teaching, learning and research: Prospects and challenges. Opara emmanuel chinonso, adalikwu mfon-ette theresa, tolorunleke caroline aduke 2023. ChatGPT for teaching, learning and research: Prospects and challenges, Glob. Acad. J. Humanit. Soc. Sci. 5 (2023).

[70] J.J. Zhu, J. Jiang, M. Yang, Z.J. Ren, ChatGPT and environmental research, Environ. Sci. Technol. (2023).

[71] N.M.S. Surameery, M.Y. Shakor, Use chat GPT to solve programming bugs, Int. J. Inf. Technol. Comput. Eng. (IJITC) (ISSN: 2455-5290) 3 (01) (2023) 17–22.

[72] F.M. Megahed, Y.J. Chen, J.A. Ferris, S. Knoth, L.A. Jones-Farmer, How generative AI models such as ChatGPT can be (Mis) used in SPC practice, education, and research? An exploratory study, 2023, arXiv preprint arXiv:2302.10916.

[73] G.H. Sun, S.H. Hoelscher, The ChatGPT storm and what faculty can do, Nurse Educ. (2023) 10–1097.

[74] C. Zhou, Q. Li, C. Li, J. Yu, Y. Liu, G. Wang ., L. Sun, A comprehensive survey on pre-trained foundation models: A history from bard to chatGPT, 2023, arXiv preprint arXiv:2302.09419.

[75] M. Dowling, B. Lucey, ChatGPT for (finance) research: The bananarama conjecture, Finance Res. Lett. (2023) 103662.

[76] A.S. George, A.H. George, A review of ChatGPT AI's impact on several business sectors, Partners Univ. Int. Innov. J. 1 (1) (2023) 9–23.

[77] J. Wang, X. Hu, W. Hou, H. Chen, R. Zheng, Y. Wang ., X. Xie, On the robustness of ChatGPT: An adversarial and out-of-distribution perspective, 2023, arXiv preprint arXiv:2302.12095.

[78] E. Costello, ChatGPT and the educational AI chatter: Full of bullshit or trying to tell us something? Postdigit. Sci. Educ. (2023) 1–6.

[79] Z. Han, F. Battaglia, A. Udaiyar, A. Fooks, S.R. Terlecky, An explorative assessment of ChatGPT as an aid in medical education: Use it with caution, MedRxiv (2023) 2023-2002.

[80] R.K. Sinha, A.D. Roy, N. Kumar, H. Mondal, R. Sinha, Applicability of ChatGPT in assisting to solve higher order problems in pathology, Cureus 15 (2) (2023).

[81] J. Gunawan, Exploring the future of nursing: Insights from the ChatGPT model, Belitung Nurs. J. 9 (1) (2023) 1–5.

[82] 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.

[83] Y. Li, Y. Duan, The performance of GPT-4 on education domain with DIKWP analysis, 2023, http://dx.doi.org/10.13140/RG.2.2.21098.39365.

[84] Y. Li, Y. Duan, The ethical performance of artificial general intelligence models based on DIKWP, 2023, http://dx.doi.org/10.13140/RG.2.2.36224.10242.

[85] S. Liu, A.P. Wright, B.L. Patterson, J.P. Wanderer, R.W. Turer, S.D. Nelson ., A. Wright, Assessing the value of ChatGPT for clinical decision support optimization, MedRxiv (2023) 2023-2002.

[86] M. Javaid, A. Haleem, R.P. Singh, ChatGPT for healthcare services: An emerging stage for an innovative perspective, BenchCouncil Trans. Benchmarks Stand. Eval. (2023) 100105.

[87] F. Antaki, S. Touma, D. Milad, J. El-Khoury, R. Duval, Evaluating the performance of chatbot in ophthalmology: An analysis of its successes and shortcomings, MedRxiv (2023) 2023-2001.

[88] S. Rana, AI and GPT for management scholars and practitioners: Guidelines and implications, FIIB Bus. Rev. 12 (1) (2023) 7–9.

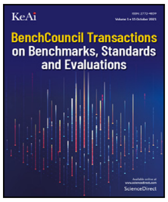
[89] R.J.M. Ventayen, OpenAI ChatGPT generated results: Similarity index of artificial intelligence-based contents, 2023, Available at SSRN 4332664.

[90] D. Sobania, M. Briesch, C. Hanna, J. Petke, An analysis of the automatic bug-fixing performance of chatbot, 2023, arXiv preprint arXiv:2301.08653.

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/



Full length article

Benchmarking HTAP databases for performance isolation and real-time analytics

Guoxin Kang [*], Simin Chen, Hongxiao Li

Institute of Computing Technology Chinese Academy of Sciences, China
University of Chinese Academy of Sciences, China

ARTICLE INFO

Keywords:
HTAP databases
Benchmark
Performance isolation
Real-time analytics

ABSTRACT

Hybrid **T**ransactional/**A**nalytical **P**rocessing (HTAP) databases are designed to execute real-time analytics and provide performance isolation for online transactions and analytical queries. Real-time analytics emphasize analyzing the fresh data generated by online transactions. And performance isolation depicts the performance interference between concurrently executing online transactions and analytical queries. However, HTAP databases are extreme lack micro-benchmarks to accurately measure data freshness. Despite the abundance of HTAP databases and benchmarks, there needs to be more thorough research on the performance isolation and real-time analytics capabilities of HTAP databases. This paper focuses on the critical designs of mainstream HTAP databases and the state-of-the-art and state-of-the-practice HTAP benchmarks. First, we systematically introduce the advanced technologies adopted by HTAP databases for real-time analytics and performance isolation capabilities. Then, we summarize the pros and cons of the state-of-the-art and state-of-the-practice HTAP benchmarks. Next, we design and implement a micro-benchmark for HTAP databases, which can precisely control the rate of fresh data generation and the granularity of fresh data access. Finally, we devise experiments to evaluate the performance isolation and real-time analytics capabilities of the state-of-the-art HTAP database. In our continued pursuit of transparency and community collaboration, we will soon make available our comprehensive specifications, meticulously crafted source code, and significant results for public access at https://www.benchcouncil.org/mOLxPBench.

1. Introduction

Hybrid **T**ransactional/**A**nalytical **P**rocessing (HTAP) databases are expected to meet the needs of real-time analytics applications [1–4] because they eliminate the extract-transform-load (ETL) processing between the OLTP database and data warehouse. HTAP databases aim to perform real-time analytics on the fresh data generated by online transactions and mitigate the performance interference between online transactions and analytical queries. To achieve the objectives mentioned above, the mainstream HTAP databases use dual data stores to guarantee performance isolation and optimize the data update propagation between the dual data stores to speed up real-time analytics.

To achieve performance isolation between online transactions and analytical queries, HTAP databases process online transactions in the row-based data store and analytical queries in the column-based data store. HTAP databases optimize the row-based and the column-based data stores, respectively, to speed the execution of online transactions and analytical queries. The row-based data store utilizes indexing and concurrency control mechanisms to facilitate update-intensive online transactions [2,5–8]. In addition, the column-based data store achieves a high compression rate and enhanced access for read-intensive analytical queries [9,10]. HTAP databases generally deploy the row-based and column-based data store on the different data nodes [11–13] to avoid high resource contention between online transactions and analytical queries. This would result in considerable latency when propagating data updates from the row-based to the column-based data store. Consequently, optimizing the data update propagation mechanism is another issue for HTAP databases to address.

Fast data update propagation from the row-based to the column-based data stores is essential for real-time analytics. The latency of data update propagation determines the freshness of the analytical data. The process of data update propagation is divided into three steps. The first step is moving the data update from the row-based to the column-based data stores. The second step is translating the row-format data into column-format data. The last step is merging the delta updates into the column-based data store. HTAP databases optimize one or all of the above steps to improve the freshness of analytical data. For example, TiDB [11] preserves only the committed change log and removes redundant information before translating it

[*] Corresponding author.
E-mail addresses: kangguoxin@ict.ac.cn (G. Kang), chensimin22z@ict.ac.cn (S. Chen), lihongxiao19@mails.ucas.ac.cn (H. Li).

https://doi.org/10.1016/j.tbench.2023.100122
Received 19 April 2023; Received in revised form 4 July 2023; Accepted 5 July 2023
Available online 8 July 2023

Table 1
The key designs of HTAP databases: can HTAP benchmarks evaluate them?

Benchmark name	Performance isolation		Real-time analytics		Component performance	
	OLTP workloads	OLAP workloads	Fresh data generation rate	Fresh data access granularity	Index mechanism	Query range control
CH-benCHmark	√	√				
HTAPBench	√	√				
CBTR	√	√				
OLxPBench	√	√				
HATtrick	√	√				
ADAPT	√	√			√	
HAP	√	√			√	
Micro-benchmark	√	√	√	√	√	√

into column-format data to decrease data movement. In contrast to TiDB, which deploys the row-based and column-based data stores on separate data nodes, some HTAP databases [9,14–17] deploy the row-based and column-based data stores on the same server to prevent data update propagation across the different data nodes. It slows down the latency of delta updates moving but poses a significant challenge to performance isolation.

Equally as important as it is to track advanced technologies for HTAP databases is to evaluate these HTAP databases. HTAP benchmarks must measure how well the HTAP databases can do performance isolation and real-time analytics. We will introduce the existing HTAP benchmarks from schema design, workload composition, and metrics as shown in Table 1.

Firstly, there are stitched schema and semantically consistent schema. The stitched schema is combined with the TPC-C schema [18] and TPC-H [19] schema. It extracts the New-Order, Stock, Customer, Orderline, Orders, Item, Warehouse, District, and History relationships from TPC-C schema [18] to integrate them with the Supplier, Country, and Region relationships of TPC-H schema [19]. CH-benCHmark [20] proposes the stitched schema, which is followed by HTAPBench [21] and Swarm64 [22]. Analytical queries cannot access the valuable data generated by online transactions and stored in the History table when using the stitched schema. And the stitched schema will affect the semantics of HTAP benchmarks. Therefore, OLxPBench [23] advocates that HTAP benchmarks should employ the semantically consistent schema instead of the stitched schema. The semantically consistent schema emphasizes that online transactions and analytical queries access the same schema. Analytical queries can access all business data generated by online transactions. The semantically consistent schema can thus reveal the performance inference between OLTP and OLAP workloads. CBTR [24, 25], OLxPBench [23], HATtrick [26], ADAPT [27], and HAP [28] benchmark all employ semantically consistent schema described in Sections 5 and 6.

Secondly, HTAP benchmarks include OLTP workloads, OLAP workloads, and hybrid workloads. OLTP workloads combine read and write operations, whereas OLAP workloads are read-intensive. Hybrid workload refers to the analytical query performed between online transactions. Existing HTAP benchmarks include OLTP and OLAP workloads to investigate performance inference between them. OLxPBench is the only benchmark that evaluates the true HTAP capability of HTAP databases using hybrid workloads. Complex online transactions and analytical queries have a lot of operations, so it is hard to judge how well each operation works on its own. ADAPT [27] and HAP [28] are Micro-benchmarks for a specific operation. However, the ADAPT and HAP benchmarks only include a handful of typical HTAP workloads. ADAPT and HAP, for instance, include an insufficient number of scan queries to evaluate index performance. Micro-benchmarks should provide point scans, small-range and large-range queries for HTAP database evaluation. There are a few Micro-benchmarks available for HTAP databases.

Thirdly, the metrics of HTAP databases are separated into two categories: throughput metrics and latency metrics. The HTAP database evaluates the throughput of OLTP workloads using the transactions per second (tps) and transactions per minute (tpmC) metrics. The HTAP database evaluates the throughput of OLAP workloads using the queries completed per second (qps) and queries completed per hour (QphH) metrics. CH-benCHmark [20] proposes the metrics $\frac{tpmC}{QphH}@tpmC$ and $\frac{tpmC}{QphH}@QphH$ for evaluating the performance isolation between OLTP and OLAP workloads. The former metric considers online transactions the primary workload, while the latter considers analytical queries the primary workload. In contrast, Anja Bog et al. [26]. establish the HATtrick benchmark, which equalizes transactional and analytical workloads. HATtrick [26] defines the throughput frontier and freshness metrics for measuring performance isolation and data freshness, as specified in Section 5.5. HTAP benchmarks utilize average latency and tail latency metrics in addition to throughput metrics. Average latency is the average time it takes for a transaction/query to be processed, whereas tail latency refers to the high percentile latency. Tail latency is an important metric to consider in HTAP databases where a small number of lengthy transactions/queries can substantially impact overall performance or user experience.

This paper makes the following contributions. (1) We systematically introduce the advanced technologies adopted by HTAP databases for these key designs; (2) We summarize the pros and cons of the state-of-the-art and state-of-the-practice HTAP benchmarks for key designs of HTAP databases; (3) We quantitatively compared the differences between micro-benchmarks and macro-benchmarks in evaluating the real-time analytical capabilities of HTAP databases. Micro-benchmark can control the generation and access granularity of fresh data, enabling precise measurement of real-time analytical capabilities of HTAP databases. (4) We measure the performance of individual components of the HTAP database, such as the indexing mechanism. By isolating specific operations, developers can test the performance of these components under different workloads and configurations, which is the foundation of component optimization.

2. Motivation — Micro-benchmarks can control the rate at which fresh data is generated and the granularity of access, which distinguishes them from macro-benchmarks

HTAP databases are extreme lack the micro-benchmark because there is no open-source micro-benchmark. We design and implement a micro-benchmark to investigate the distinction between the micro-benchmark and the macro-benchmark. We select the state-of-the-art HTAP benchmark OLxPBench as the micro-benchmark comparison object. Micro-benchmark is better suited for real-time analytics evaluation because it precisely controls the rate at which fresh data is generated and the granularity of fresh data access. Micro-benchmark queries typically consist of a single statement. For instance, the analytical query calculates the number of rows within a specified range. This indicates that the computational intensity of analytical queries can be managed by adjusting their computational range. And the transactional query updates the value of the specified column in a random row.

Micro-benchmark can adjust the rate at which fresh data is generated to assess the performance of data update propagation between the transactional and analytical instances. The performance interference between transactional and analytical queries can be disregarded when

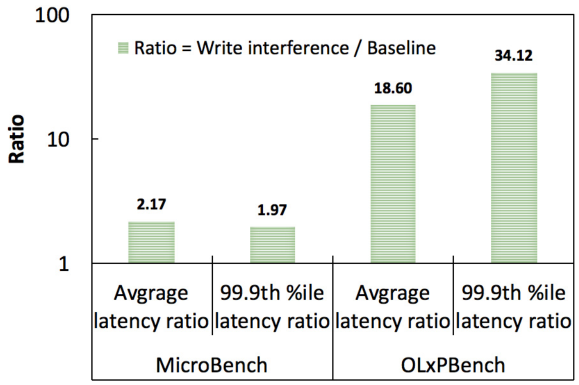


Fig. 1. This figure reveals that the micro-benchmark can accurately measure the real-time analytical capabilities of the HTAP database by controlling read and write interference.

the number of concurrent requests is low. Consequently, almost all of the growing proportion of analytical latency is due to the propagation of data updates. The online transactions and analytical queries in OLxPBench are too complex to control the write and read ranges precisely. The New-Order transaction, for instance, involves numerous inserting and updating operations. The analytical query (Q6) includes operations involving aggregation and sub-selection. This causes the New-Order transaction to generate fresh data that is only partially required by the analytical query (Q6). However, the analytical query must wait for all data updates to propagate before accessing the fresh data. Unlike OLxPBench, micro-benchmark makes it simple to control the rate at which fresh data is generated and the granularity of access to analytical queries on fresh data.

Fig. 1 compares the impact of simple write operations and the New-Order transaction on the measurement of data freshness. The New-Order transaction includes an excessive number of updating and inserting operations, thereby introducing data synchronization that is unnecessary for measuring data freshness. The greater the ratio, the more data needs to be synchronized. It demonstrates that the tail latency of analytical queries (Baseline) increases approximately one-fold when the micro-benchmark is used to simulate write interference. The New-Order transaction contains numerous inserting and updating operations, so the tail latency of the baseline (Q6) increases by more than 36 times when OLxPBench is used. The greater the number of inserting and updating operations, the greater the number of data updates that must be synchronized between transactional and analytical instances. However, not all data updates resulting from online transactions are required for analytical queries. The data freshness measurement will be affected by data updates that are not required by the analytical query. Measuring data freshness requires precise control over the rate of fresh data generation and access granularity.

3. Key designs of mainstream HTAP databases

The mainstream HTAP databases are designed for two objects: real-time analytics and performance isolation. Performance isolation emphasizes that online transactions and analytical queries execute concurrently without affecting each other's performance. Real-time analytics means analyzing the fresh data generated by online transactions as soon as possible. Online transactions and analytical queries can achieve superior isolation performance through the use of independent storage engines. However, the necessity of data synchronization between row-based and column-based storage engines undeniably introduces data synchronization latency. Consequently, real-time access to fresh data during analytical queries becomes a formidable challenge. Therefore, it is challenging for HTAP databases to provide real-time analytics and performance isolation capabilities. Some HTAP databases deploy the transactional and analytical instances on the same server to avoid long turnaround times for delta updates. And other HTAP databases handle online transactions and analytical queries on separate servers to prevent performance interference. This section studies how HTAP databases accomplish real-time analytics and performance isolation.

3.1. Performance isolation

Single-node HTAP databases implement row-based data storage for online transactions and column-based data storage for analytical queries. Because of the intense resource contention, single-node HTAP databases cannot provide performance isolation [29]. Previous works [30,31] have proposed various approaches to partitioned hardware resources to ensure performance isolation. Raza et al. [30] divide the CPU and memory resources into two groups: the first group binds with the specified transactional instance and analytical instance. In contrast, the second group comprises reserved resources assigned based on actual requirements. By dividing the last-level cache (LLC) between the analytical queries and the online transactions, Sirin et al. [31] reduce the performance impact of the analytical queries on the online transactions. Polynesia [15] identifies the root cause of performance interference as the sharing of hardware resources and consequently provides an isolated computing resource for online transactions and analytical queries.

Distributed HTAP databases [11–13] deploy row-based and column-based data stores on separate servers, thereby wholly resolving the issue of resource contention. TiDB [11] implements the $TiKV$ and $TiFlash$ instances for row-based and column-based data stores, respectively. It employs the raft algorithm to replicate asynchronously $TiKV$ logs to $TiFlash$ instances. Due to the collaborative capabilities of Google's internal systems, F1 Lightning [12] contributes a loosely coupled HTAP solution that enables the $F1$ $Query$ $engine$ to function with existing OLTP systems and data sources [32–37]. As a result, F1 Lightning [12] need to utilize the $Lightning$ component to capture the data updates from various data sources and translate them into the unified format data.

SingleStore [13] and OceanBase [38] are well-known distributed HTAP databases. They all utilize unified storage to facilitate online transactions and analytical queries. OceanBase [38] demonstrates commendable proficiency in resource isolation. PolarDB-IMCI [39] also provides effective resource isolation for transactional and analytical queries.

3.2. Real-time analytics

Initially, HTAP databases deploy the transactional and analytical instances on a single server to obtain the fresh data generated by online transactions. SAP HANA [9,40] maintains multiple delta update stores for the same table, allowing online transactions updating and existing data updates merging process to be performed in different delta stores. It is permitted for analytical queries to simultaneously access the freshest data in multiple deltas and column-based data stores. SAP HANA [9,40], DB2 BLU [17] and Oracle [14] have implemented an in-memory column-based data store for fast analytics. DB2 [17] supports HTAP workloads with BLU acceleration. Oracle [14] and DB2 BLU [17] make use of numerous analytical optimization technologies, including compression and single-instruction multiple-data (SIMD). It cannot update column data in real-time because data updates are only merged to the column-based data store when the ratio of data updates exceeds a certain threshold.

With the growing amount of real-time data, the single-node HTAP database cannot meet the high scalability and availability requirements. Oracle, for instance, releases a new distributed version that provides scale-out compute and storage resources and implements a real-time column-based data duplication mechanism for high availability requirements [41]. And then comes the issue of data update propagation. Because data updates must propagate from transactional servers to analytical servers, it is challenging for distributed HTAP databases to guarantee that analytical queries can access the most recent data updates [42]. Both TiDB and F1 Lightning have specialized components for data update propagation. TiDB [11] utilizes the $Logreplication$ process asynchronously to maintain data consistency between $TiKV$ and $TiFlash$ instances. The $Changepump$ component of F1 Lighting [12] provides a consistency protocol that enables analytical queries to access in-memory delta updates generated by online transactions immediately. Once a system failure occurs, the in-memory delta updates are recoverable through the transactional log.

4. Can existing HTAP benchmarks evaluate the key design of HTAP databases?

HTAP databases provide performance isolation for OLTP workloads and OLAP workloads while ensuring that OLAP workloads have access to fresh data generated by OLTP workloads. Evaluating the performance of individual HTAP database components is paramount in optimizing their efficiency.

Table 1 summarizes the existing HTAP benchmarks. Currently, open-source HTAP benchmarks are macro-benchmarks, encompassing intricate online transactions and analytical queries. This approach facilitates a comprehensive assessment of the performance isolation capabilities inherent in HTAP databases. However, due to the extensive number of statements within online transactions and analytical queries, accurately evaluating the performance of specific HTAP database components, such as index performance, presents a considerable challenge. Benchmarking can utilize range scan queries to measure the performance of various index mechanisms in HTAP databases. For instance, theoretically, point queries can effectively evaluate the performance of Hash indexes and LSM-Tree indexes. In an ideal scenario, a Hash index only needs to perform a single hash computation on the primary key to find the corresponding record, while an LSM-Tree index, using a binary search algorithm, requires multi-level searching. However, since Hash indexes are unordered, handling range queries necessitates scanning the entire index space, in contrast, the ordered LSM-Tree indexes perform more efficiently during range queries. Hence, range scan queries can effectively test and compare the performance of different index mechanisms. In this research, the impact of query scope on computational workload intensity is meticulously investigated. By strategically manipulating the scope of a query, it is possible to exert greater control over the computational demands of the workload. A comparative analysis is performed, examining point queries, small-range queries, and large-range queries, all with identical transmission rates. The results reveal distinct discrepancies in the required computational resources and the subsequent performance outcomes for each query type. This study offers valuable insights into optimizing query execution and enhancing system performance.

Concurrently, the complexity of workloads makes it arduous to regulate the generation rate and access the granularity of fresh data. This predicament leads to measurement biases concerning data freshness. Section 2 elucidates the distinctions between macro-benchmarks and micro-benchmarks in the context of controlling fresh data. Micro-benchmarks are indispensable for appraising the real-time analysis capabilities of HTAP databases. Despite their importance, there is a notable absence of open-source micro-benchmarks explicitly designed for HTAP databases. Consequently, there is an urgent need within the industry and academia to develop tailor-made micro-benchmarks specifically intended for HTAP databases.

5. Macro-benchmarks for HTAP databases

5.1. CH-benchmark

5.1.1. Schema design

CH-benCHmark [20] combines the TPC-C [18] and TPC-H [19] schema. The CH-benCHmark schema retains all TPC-C tables and adds the Supplier, Nation, and Region tables of TPC-H. Fig. 2 depicts the relationships between nine tables.

5.1.2. Workload description

CH-benCHmark provides both online transactions and analytical queries. The OLTP workloads are the same as the TPC-C transactions which are New-Order, Payment, Order-Status, Delivery, and Stock-Level transactions. The default percentages for the aforementioned five transactions are 44%, 44%, 4%, 4%, and 4%, respectively. Order-Status and Stock-Level are read-only transactions, and the remaining three are update-intensive transactions. The 22 analytical queries in CH-benCHmark are derived from the TPC-H benchmark. Analytical queries retain the majority of business semantics but make adjustments based on the CH-benCHmark schema.

5.1.3. Evaluation and metrics

CH-benCHmark evaluates the OLTP, OLAP, and mixed performance of HTAP databases. It regulates the rate at which online transactions and analytical queries are sent by setting the number of request-sending threads. It measures the performance of HTAP databases using response time and throughput metrics. The transactions per minute (tpmC) metric is utilized to measure the throughput of OLTP workloads. And the queries per hour (QphH) metric is utilized to measure the throughput of OLAP workloads. CH-benCHmark inventively designs the $\frac{tpmC}{QphH}@tpmC$ and $\frac{tpmC}{QphH}@QphH$ metrics to measure the performance of mixed workloads that consist of both OLTP and OLAP workloads. $@tpmC$ indicates OLTP as the dominant workload, whereas $@QphH$ indicates OLAP as the dominant workload.

For instance, as shown in expression (1), when transactional and analytical workloads are executed sequentially, their respective throughputs are assumed to be 5084 tpmC and 895.6 QphH.

$$P_1(OLTP) = \frac{5084tpmC}{895.6QphH}@5084tpmC. \tag{1}$$

As shown in expression (2), when executing mixed workloads concurrently, the OLTP and OLAP throughputs are assumed to be 5188 tpmC and 804.2 QphH, respectively.

$$P_2(OLTP) = \frac{5188tpmC}{804.2QphH}@5188tpmC. \tag{2}$$

$P_1(OLTP)$ equals $5.7@5084tpmC$, which is less than 6.5@ 5188$tpmC$ of $P_2(OLTP)$. The results indicate that analytical queries do not hinder the performance of online transactions in this experiment. In addition, CH-benCHmark is the first HTAP benchmark that defines data freshness. CH-benchmark decides whether to use the most recent data for analytical queries by setting either a time threshold or a number of transactions.

5.2. HTAPBench

HTAPBench [21] adopts the same schema as CH-benCHmark, which integrates TPC-C and TPC-H schema. HTAPBench takes five online transactions from TPC-C and 22 analytical queries from TPC-H.

The most distinct aspect between HTAPBench and CH-benCHmark is that HTAPBench proposes a unified metric for HTAP databases, as shown in the expression (3).

$$QpHpW = \frac{QphH}{\#OLAPworkers}@tpmC. \tag{3}$$

QpHpW represents the analytical queries completed in an hour per analytical worker. When the throughput of online transactions remains

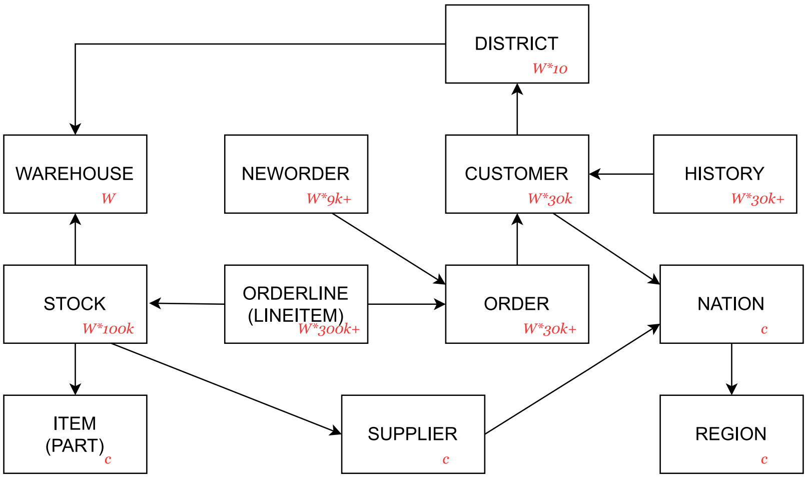


Fig. 2. The schema of CH-benCHmark.

constant, the greater the number of analytical queries completed per hour per analytical worker, the better the performance of the HTAP database.

5.3. CBTR

CBTR [24,25] is the first HTAP benchmark that adopts the semantically consistent schema. The semantically consistent schema, unlike the stitched schema, allows OLTP and OLAP workloads to operate on the same tables as opposed to separate tables for each workload. The schema of CBTR includes 18 tables, which are extracted from the real-world order-to-cash scenario. The normalization of CBTR's schema is configurable, with 1NF being the default. Different normalization levels produce varying degrees of data redundancy, which has a direct impact on the total number of columns. For instance, the schema with the highest degree of redundancy contains 2316 columns in total.

CBTR provides four online read-update transactions, three online read-only transactions, and four online analytical queries. CBTR utilizes data from actual business scenarios rather than synthetic data generated by data generators. However, CBTR is not widely recognized due to its closed-source nature.

5.4. OLxPBench

5.4.1. Schema design

The OLxPBench suite [23] proposes creatively that HTAP benchmarks necessitate a semantically consistent schema. Semantically consistent schema emphasizes that online transactions and analytical queries should use the same data. There are three benchmarks in the OLxPBench suite: subenchmark for general scenarios, fibenchmark for financial scenarios, and tabenchmark for telecom scenarios. Subenchmark reuses the schema of the TPC-C [18], which consists of nine tables. The schema of fibenchmark is derived from that of SmallBank benchmark [43] and has three tables: $ACCOUNT$, $SAVING$, and $CHECKING$ tables. TATP [44], which has four tables, including $SUBSCRIBER$, $SPECIAL\ FACILITY$, $ACCESS\ INFO$, and $CALL\ FORWARDING$ tables, is the source of inspiration for tabenchmark. Tabenchmark modifies the SUBSCRIBER table by expanding a composite primary key.

5.4.2. Workload description

The OLxPBench benchmark suite consists of 18 online transactions, 18 analytical queries, and 17 hybrid transactions. The online transactions of the original OLTP benchmarks remain the same. Just 8% of online transactions in Subenchmark are read-only. 15% of online transactions in Fibenchmark are read-only. 80% of online transactions in Tabenchmark are read-only. In addition, it increases the analytical queries and hybrid transactions based on the semantically consistent schema. The analytical queries consist of complicated analytical statements like aggregation and multi-join. The hybrid transaction incorporates an analytical statement into an online transaction. Read-only hybrid transactions make up 60%, 20%, and 40% of the subenchmark, fibenchmark, and tabenchmark, respectively.

5.4.3. Evaluation and metrics

OLxPBench suite evaluates the performance isolation between the online transactions and analytical queries. It begins by determining the peak throughput of online transactions and analytical queries. Fix the request send rate for online transactions or analytical queries x_f, and progressively increase the request send rate for the other instances x_i. If the throughput and latency of x_f vary minimally, there is no performance interference between online transactions and analytical queries; contrarily, the higher the fluctuation in performance of x_f, the greater the performance interference between online transactions and analytical queries. In addition, the OLxPBench suite evaluates the HTAP performance of HTAP databases using hybrid transactions. Moreover, the scalability of HTAP databases is evaluated.

5.5. HATtrick

5.5.1. Schema design

The schema of the HATtrick benchmark [26] is modified based on Star-Schema Benchmark (SSB) [45]. The schema of the HATtrick benchmark newly adds the HISTORY and FRESHNESS table and appends new attributes to the CUSTOMER, SUPPLIER, and PART table The schema consists of seven tables, as shown in Fig. 3.

5.5.2. Workload description

HATtrick includes both online transactions and analytical queries. It offers three online transactions comparable to the TPC-C benchmark. The transactional workloads consist of 48 percent New-Order, 48 percent Payment, and 4 percent Count orders. The New-order and Payment transactions are update-intensive, while the Count orders transaction is read-only.

Thirteen analytical queries are derived from the SSB benchmark and modified slightly to conform with the schema.

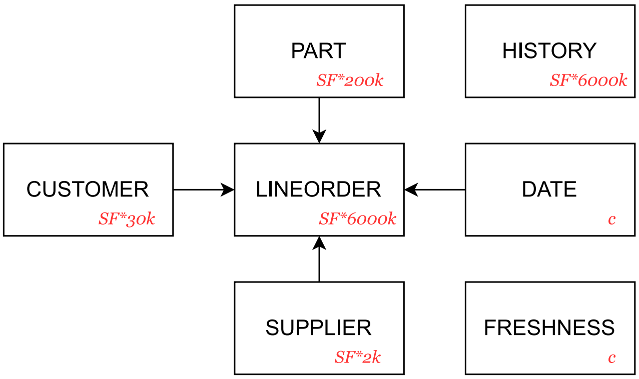


Fig. 3. The schema of SSB.

5.5.3. Evaluation and metrics

HATtrick proposes the throughput frontier and freshness score metrics to measure the performance isolation and data freshness of HTAP databases. The throughput frontier is visualized as a curve with the transactional throughput x_t on the horizontal axis and the analytical throughput y_a on the vertical axis. The maximum transactional throughput is X_t, and the maximum analytical throughput is Y_a. The line formed by the coordinates $(0, Y_a)$, (X_t, Y_a), and $(X_t, 0)$ is the bounding line. And the line formed by the coordinates $(0, Y_a)$ and $(X_t, 0)$ is the proportional line. If the throughput frontier is close to the bounding line, HTAP database performance isolation is stable. If the throughput frontier is below the proportional line, it indicates a significant performance interference between online transactions and analytical queries. The freshness score metric refers to the delay when analytical queries can access the latest data generated by online transactions.

5.6. Advantages and disadvantages

Evaluations of HTAP databases require the proper schema, workloads, and metrics. Ch-benCHmark and HTAPBench are the first HTAP benchmarks to implement the stitched schema, separated online transactions and analytical queries, and metrics described in Section 5.1. CBTR, OLxPBench suite, and HATtrick implement a semantically consistent schema to analyze the performance isolation between online transactions and analytical queries. The CBTR schema is derived from the actual production environment. The OLxPBench suite implements domain-specific benchmarks and innovative hybrid transactions. HATtrick contributes the throughput frontier and freshness score metrics.

6. Micro-benchmarks for HTAP databases

6.1. ADAPT

ADAPT [27] is a synthetic benchmark that extracts typical operations from the production environment [46]. The schema contains both narrow and wide tables. The narrow table contains 50 columns, and the wide table contains 500 columns. ADAPT benchmark contributes five queries: insert query, scan query, maximum aggregate query, sum aggregate query, and join query. ADAPT benchmark lacks delete, update, and point scan queries.

6.2. HAP

Based on the ADAPT benchmark, the HAP benchmark [28] reduces the number of columns in narrow and wide tables. The narrow table has 16 columns, whereas the wide table has 160 columns. HAT benchmark contains six queries: point query, count aggregate query, sum aggregate query, insert query, delete query, and update query. The delete, update, and point scan queries have recently been added to the HAP benchmark. At the same time, it deletes the scan and the join queries.

6.3. Advantages and disadvantages

The ADAPT [27] and HAP [28] benchmarks abstract the basic HTAP operations. However, they contain a limited number of typical HTAP workloads and are not open-source. The micro-benchmark should provide a variety of scan queries, including point queries, small-range queries, and large-range queries. The variety of scan queries is crucial for the index optimization of HTAP databases. In addition, micro-benchmarks must ensure that the read and write operations access the same columns to evaluate the date update propagation capability.

7. Micro-benchmark

7.1. Range setting method

Informed by a theoretical framework, the parameters of a scan query range are thoughtfully established. We start by supposing that the total count of records in a table is represented as S. The range for this operation is defined between two integer values, a lower bound L, and an upper bound U. As described in Eqs. (4) and (5), the boundaries of L and U are established with the essential stipulation that L must always remain less than U. The desired range for our scan query is the calculated difference between U and L. Our ultimate aim is to determine the average range, and then to utilize this average as a pivotal point. Upon establishing this pivotal point, we then select scatter points of several orders of magnitude beneath it to determine the scan query range.

A key strategy in achieving this objective involves a recalibration of the range values, effecting a transformation into a summation of multiple terms by increasing them by 1. This process is illustrated in Eq. (6). Following this, we examine the pattern of the data, accumulate the range values, and then divide this sum by the number of range values. This leads us to the determination of the average range, as represented in Eq. (7). Our calculations reveal that the proximate value of the average range in this random configuration is equivalent to one-third of S, as demonstrated in Eq. (8). Based on these findings, we establish the range of the scan query to fall within the parameters of 0.5% and 10% of the total record count, S. Aggregate and scan queries exhibit the capacity to meticulously regulate the granularity of access to fresh data by expertly delineating the scope of the inquiry. This stands as a distinguishing hallmark, setting micro-benchmarks apart from conventional HTAP benchmarks.

$$L \in [1, S-2] \cap \mathbb{Z}. \tag{4}$$

$$U \in [L+1, S] \cap \mathbb{Z}. \tag{5}$$

$$\begin{aligned} avgScanSize &= \frac{-1 + \sum_{x=1}^{S-1} x(S-x)}{-1 + \sum_{x=1}^{S-1} x}, \quad (6)\\ &= \frac{\frac{1}{6}(S-1)S(S+1) - 1}{\frac{1}{2}(S-1)S - 1}, \quad (7)\\ &\approx \frac{1}{3}S. \quad (8) \end{aligned}$$

7.2. The design and implementation

There is no open-source micro-benchmark for HTAP databases. The micro-benchmark could precisely regulate the read/write ratio for a comprehensive evaluation of HTAP databases. Therefore, we mimic the ADAPT and HAP benchmarks to design and implement the micro-benchmark, which accomplishes the six queries listed below. Moreover, the micro-benchmark contains a single table with 59 attributes named $ITEM$. The attributes in the $ITEM$ table are derived from the actual e-commerce applications.

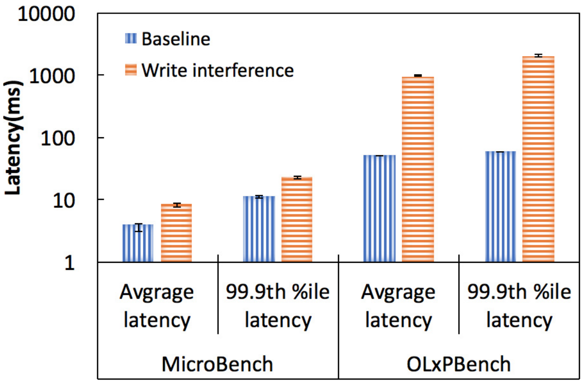


Fig. 4. This figure illustrates the distinction between the micro-benchmark and OLxPBench, the state-of-the-art HTAP benchmark.

Q1 is a point-get query retrieves the record where the primary key equals a random number. Q4 is a small-range scan query that randomly retrieves 0.5% of the records. Q5 is a large-range scan query that randomly retrieves 10% of the records. Q3 is an update query that updates the specific value of a random record. HTAP database indexing and writing speeds can be measured with Q1, Q4, Q5, and Q3. Q2 is an aggregate query that counts the records in a random range. Q6 is a small-range aggregate query that counts 0.5% of the records. Q2 and Q6 are useful to measure the OLAP performance of HTAP databases. All experimental results reported in this paper are the mean and standard deviation of five independent runs.

Q1: **SELECT** i_1, i_2, ... ,i_k **FROM** ITEM
WHERE i_{id} = v;

Q2: **SELECT COUNT**(∗) **FROM** ITEM
WHERE $i_{id} \in [v_s, v_e]$;

Q3: **UPDATE** ITEM **SET** $i_r = v_r$
WHERE $i_{id} = v_s$;

Q4: **SELECT** i_1, i_2, ... ,i_k **FROM** ITEM
WHERE $i_{id} \in [v_s, v_p]$;

Q5: **SELECT** i_1, i_2, ... ,i_k **FROM** ITEM
WHERE $i_{id} \in [v_s, v_q]$;

Q6: **SELECT COUNT**(∗) **FROM** ITEM
WHERE $i_{id} \in [v_s, v_q]$;

8. Evaluation

8.1. Experimental setup

8.1.1. Environment

The server node has two Intel Xeon E5-2699v4@2.20 GHz CPUs, 128 GB memory, and two 2TB SSD. The client node has two Intel Xeon E5645@2.40 GHz CPUs, 48 GB memory, and eight 2TB HDDs. The server and client run on Ubuntu 20.04 version and are connected by a 10 Gbps Ethernet network.

8.1.2. Database

TiDB is an industry-standard HTAP database, and its version is 6.1.0. We deploy the *tidb* instance, the *TiKV* instance, and the *TiFlash* instance on the same server in order to evaluate the real-time analytics and performance isolation capabilities in depth. The components of TiDB are described in detail in Section 3.

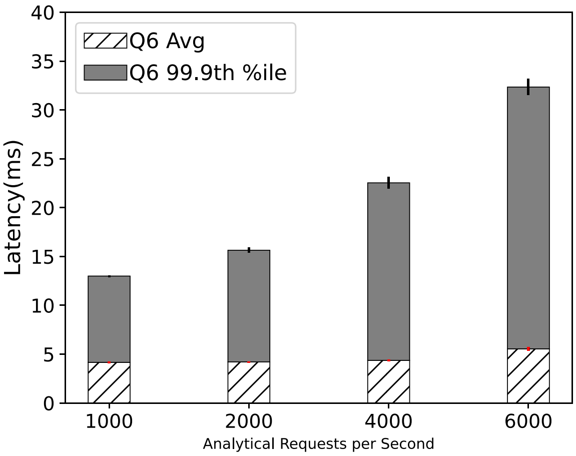


Fig. 5. Small aggregate query performance.

8.2. Comparing micro-benchmark to the state-of-the-art HTAP benchmark

As the experimental workload, we selected the New-Order transaction and analytical query (Q6) from the Subenchmark in the OLxPBench suite. In addition, we utilize Q2 and Q3 as experimental workloads. Each experiment is performed five times independently, and the mean and standard deviation are reported. To avoid performance interference between transactional and analytical instances, concurrent requests are limited to 100 transactional and analytical requests per second. As depicted in Fig. 4, the average and tail latency of Q2 nearly doubles with interference from Q3. Due to the interference of the New-Order transaction, the average and tail latency of Q6 increased by 19 and 34 times, respectively. This is because the New-Order transaction in OLxPBench contains an excessive number of inserting and updating operations, resulting in an excessive number of data updates to propagate. However, not all New-Order data update records are required by the analytical query (Q6). In micro-benchmark, Q2 requires all data updates produced by Q3. Unnecessary data updates introduce excessive synchronization latency, resulting in inaccurate data freshness measurements. The premise of measuring data freshness is therefore to strictly control the generation rate and gain access to the granularity of fresh data.

8.3. Scan performance

We set up the point query, small range query, and large range query to fully evaluate the index performance of HTAP databases. The scan performance is shown in Fig. 9, Fig. 11, and Fig. 10. The diagonal area represents the average latency, while the area with the gray shading represents the tail latency. Every experiment is shown in this manner and will not be discussed below. Peak throughput for point query, small-range scan query, and large-range scan query exceed 20,000, 10,000, and 400 tps, respectively. Peak throughput decreases as the total of scan records expands. The average latency of the point query illustrated in Fig. 9 is less than five milliseconds. The average latency of the small-range scan query illustrated in Fig. 11 is less than ten milliseconds. *TiKV* has implemented a scalable, ordered LSM-Tree index, with experimental results demonstrating that the latency for both point queries and small-range scan queries is within the millisecond level. When handling point queries, TiDB performs binary searches based on the primary key's value, necessitating multi-level searching to locate the relevant data block. Furthermore, TiDB has parallel optimizations for range queries, using a Coprocessor to concurrently access ordered blocks, thereby accelerating the processing speed for range queries.

The average latency of the large-range scan query illustrated in Fig. 10 is the greatest and exceeds twenty milliseconds. And the 99.9th

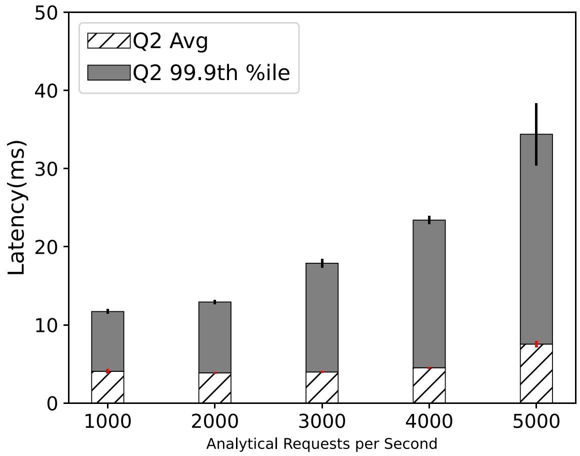


Fig. 6. Random aggregate query performance.

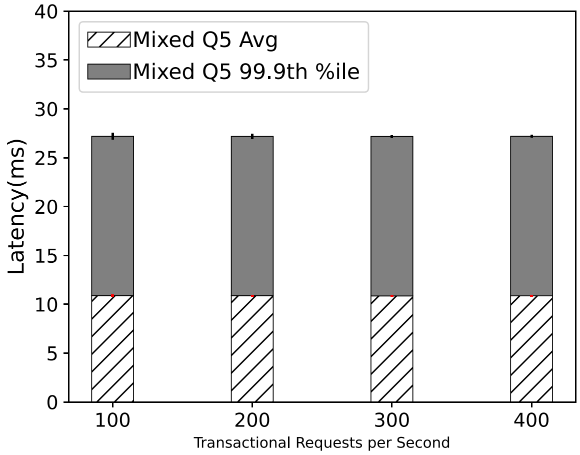


Fig. 7. Performance interference of Q5 on Q2.

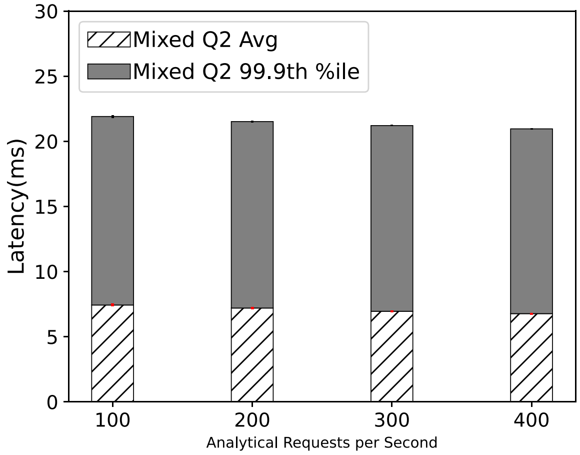


Fig. 8. Performance interference of Q2 on Q5.

percentile latency of the large-range scan query is greater than 45 ms. The point query retrieves the targeted record by primary key. The range queries push the task down to $TiKV$ instance execution and summarize the $TiKV$ instance return results in the SQL engine. In addition, range scan queries retrieve the continuous records stored in a small number of regions, which can lead to access hotspot issues.

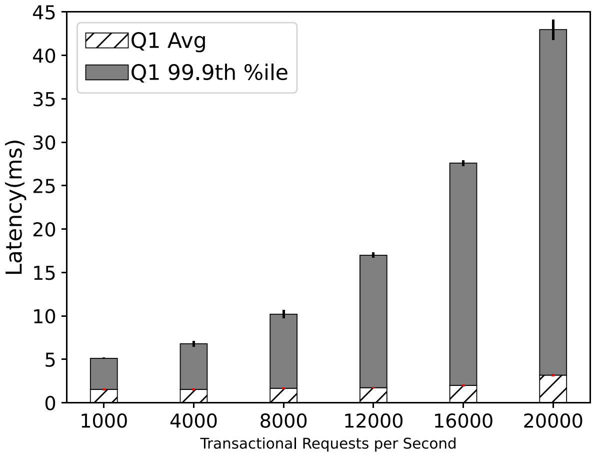


Fig. 9. Point-get query performance.

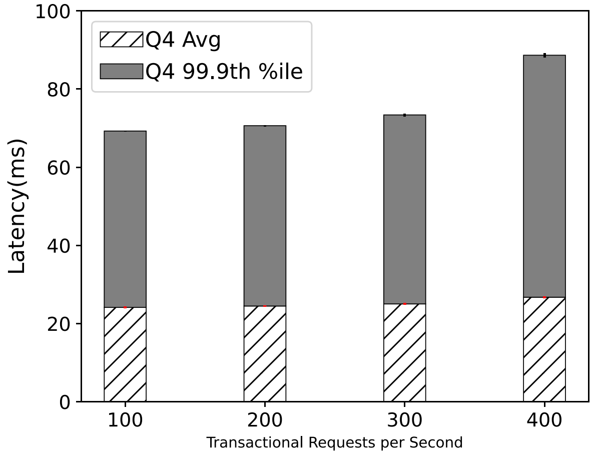


Fig. 10. Large-range scan query performance.

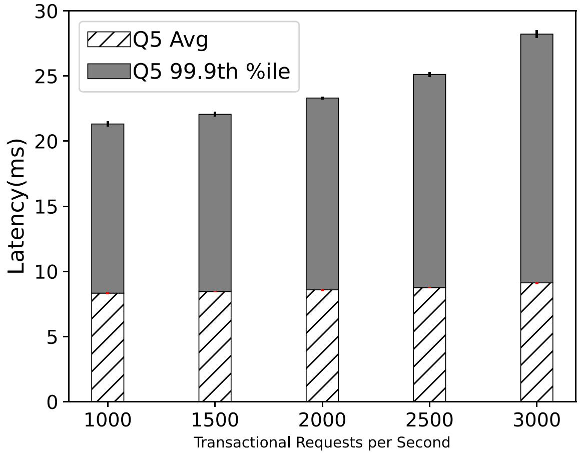


Fig. 11. Small-range scan query performance.

8.4. Update performance

We use the update queries that follow the uniform distribution to measure the update performance. The performance results of the update operation are depicted in Fig. 12. The average latency increases 1.9× with the transactional requests increasing from 1000 tps to 10000 tps. Meanwhile, the 99.9th percentile latency increases from 10.4 ms

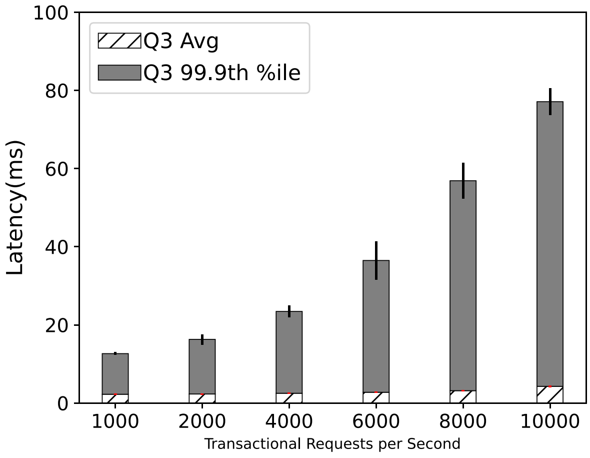


Fig. 12. Update query performance.

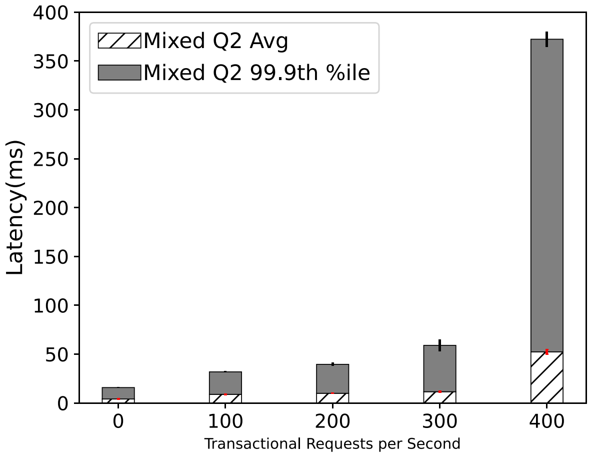


Fig. 13. Real-time analytic performance.

to 72.8 ms. In the experiment, write conflicts gradually appear as the number of concurrent update requests increases. The reason for this experimental phenomenon is that TiDB uses the pessimistic model by default, and the "autocommit" option is enabled. A transaction is initially committed as an optimistic transaction and then, if a write conflict occurs, as a pessimistic transaction. If there are violent write conflicts, it is recommended to disable "autocommit" option.

8.5. Aggregate performance

Q2 and Q6 return the number of rows retrieved. When analytical requests per second are less than 4000 tps, the average latency of Q2 and Q6 is around four milliseconds. As shown in Figs. 5 and 6, the maximum 99.9th percentile latency for Q2 is 34.39 ms, and that for Q6 is 26.79 ms. Q6 has a greater peak throughput and a shorter tail latency than Q2 due to the fact that Q2 requires more calculations. For data aggregation, TiDB implements the hash aggregation operator and the stream aggregation operator. The SQL engine uses the stream aggregation operation to deal with the COUNT(*) function. The stream aggregation operator requires less memory than the stream aggregation operator.

8.6. Hybrid performance

8.6.1. Performance isolation evaluation

To investigate the performance isolation issue, we deploy the $TiKV$ and $TiFlash$ instances on the same server. We employ read-only queries for the performance isolation evaluation to prevent the interference of data update propagation. $TableRangeScan$ and $StreamAgg$ are the operators utilized for the large-range scan and aggregate queries, respectively. The send rate of the Q5 remains constant while the sending rate of the aggregate inquiries steadily increases from 100 tps to 400 tps in the first set of experiments, as shown in Fig. 7. The send rate of the Q2 remains constant in the second set of experiments, as shown in Fig. 8, while the sending rate of the scan queries steadily increases from 100 tps to 400 tps. As the sending rate of interference inquiries rises, the latency of scan and aggregate queries remains relatively constant within the error bounds. Even when placed on the same server, there is not much performance interference between $TiKV$ and $TiFlash$ instances when there is no resource competition between mixed workloads.

8.6.2. Real-time analytic evaluation

We deploy the $TiKV$ and $TiFlash$ instances in the same server to guarantee the analytical query analyzes the fresh transactional data as soon as possible. To minimize interference, the $TiKV$ and $TiFlash$ instances are deployed on the different solid-state drives. We keep the analytical requests per second constant, increasing the proportion of updated data to guarantee an increasing proportion of fresh data that analytical requests can access. As shown in Fig. 13, the number of update requests per second rises from 100 to 400 tps. The send rate of the analytical queries remains constant, and we collect the analytical queries' latency results. A low number of concurrent requests avoids performance interference between update and aggregate queries, which is demonstrated in Section 8.6.1. Both the average and 99.9th percentile latency rise by more than a factor of one due to the propagation of data updates. The average latency of analytical queries is 52.26 ms, and the 99.9 percentile delay is 320.04 ms when the send rate of update queries is 400 tps. The aforementioned performance results indicate that TiDB can complete real-time analytics in 500 ms without transferring data updates across nodes.

9. Conclusion

This paper involves a thorough introduction of HTAP database strategies for enhancing performance isolation and real-time analytics. In addition, we compare state-of-the-art and best-practice HTAP benchmarks in terms of schema model, workloads, and evaluation metrics. The CBTR, OLxPBench, HATtrick, ADAPT, and HAP benchmarks all use the semantically consistent schema. OLxPBench is innovative and provides a hybrid transaction that executes the analytical statement between the online transaction. And HATtrick contributes the throughput frontier and freshness metrics.

Currently, HTAP databases are severely lacking in micro-benchmarks to precisely manage read and write ranges. Consequently, we implement a micro-benchmark in Section 7 to measure the performance isolation and real-time analytics capabilities of HTAP databases. When the number of concurrent requests is modest, the performance of transactional and analytical instances on the same server does not interfere with one another. The propagation of data updates on the same node promotes the preservation of analytical data's freshness. Moreover, rigorous resource partitioning between transactional and analytical instances may facilitate dual-format HTAP databases to support both performance isolation and real-time analytics.

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] Ronald Barber, Christian Garcia-Arellano, Ronen Grosman, Rene Mueller, Vijayshankar Raman, Richard Sidle, Matt Spilchen, Adam J. Storm, Yuanyuan Tian, Pinar Tözün, et al., Evolving databases for new-gen big data applications, in: CIDR, 2017.

[2] Chen Luo, Pinar Tözün, Yuanyuan Tian, Ronald Barber, Vijayshankar Raman, Richard Sidle, Umzi: Unified multi-zone indexing for large-scale HTAP, in: Advances in Database Technology-22nd International Conference on Extending Database Technology, EDBT 2019, Lisbon, Portugal, March 26–29, 2019, OpenProceedings. org, 2019, pp. 1–12.

[3] Fatma Özcan, Yuanyuan Tian, Pinar Tözün, Hybrid transactional/analytical processing: A survey, in: Proceedings of the 2017 ACM International Conference on Management of Data, 2017, pp. 1771–1775.

[4] Hemant Saxena, Lukasz Golab, Stratos Idreos, Ihab F. Ilyas, Real-time LSM-trees for HTAP workloads, 2021, arXiv preprint arXiv:2101.06801.

[5] Yihan Sun, Guy E. Blelloch, Wan Shen Lim, Andrew Pavlo, On supporting efficient snapshot isolation for hybrid workloads with multi-versioned indexes, Proc. VLDB Endow. 13 (2) (2019).

[6] R. Malinga Perera, Bastian Oetomo, Benjamin I.P. Rubinstein, Renata Borovica-Gajic, No DBA? No regret! multi-armed bandits for index tuning of analytical and HTAP workloads with provable guarantees, 2021, arXiv preprint arXiv:2108.10130.

[7] Jinwei Guo, Peng Cai, Jiahao Wang, Weining Qian, Aoying Zhou, Adaptive optimistic concurrency control for heterogeneous workloads, Proc. VLDB Endow. 12 (5) (2019) 584–596.

[8] Tobias Vinçon, Christian Knödler, Leonardo Solis-Vasquez, Arthur Bernhardt, Sajjad Tamimi, Lukas Weber, Florian Stock, Andreas Koch, Ilia Petrov, Near-data processing in database systems on native computational storage under htap workloads, Proc. VLDB Endow. 15 (10) (2022) 1991–2004.

[9] Franz Färber, Sang Kyun Cha, Jürgen Primsch, Christof Bornhövd, Stefan Sigg, Wolfgang Lehner, SAP HANA database: data management for modern business applications, ACM SIGMOD Rec. 40 (4) (2012) 45–51.

[10] Michael Abebe, Horatiu Lazu, Khuzaima Daudjee, Proteus: Autonomous adaptive storage for mixed workloads, in: Proceedings of the 2022 International Conference on Management of Data, 2022, pp. 700–714.

[11] Dongxu Huang, Qi Liu, Qiu Cui, Zhuhe Fang, Xiaoyu Ma, Fei Xu, Li Shen, Liu Tang, Yuxing Zhou, Menglong Huang, et al., TiDB: a Raft-based HTAP database, Proc. VLDB Endow. 13 (12) (2020) 3072–3084.

[12] Jiacheng Yang, Ian Rae, Jun Xu, Jeff Shute, Zhan Yuan, Kelvin Lau, Qiang Zeng, Xi Zhao, Jun Ma, Ziyang Chen, et al., F1 lightning: HTAP as a service, Proc. VLDB Endow. 13 (12) (2020) 3313–3325.

[13] Adam Prout, Szu-Po Wang, Joseph Victor, Zhou Sun, Yongzhu Li, Jack Chen, Evan Bergeron, Eric Hanson, Robert Walzer, Rodrigo Gomes, et al., Cloud-native transactions and analytics in SingleStore, in: Proceedings of the 2022 International Conference on Management of Data, 2022, pp. 2340–2352.

[14] Tirthankar Lahiri, Shasank Chavan, Maria Colgan, Dinesh Das, Amit Ganesh, Mike Gleeson, Sanket Hase, Allison Holloway, Jesse Kamp, Teck-Hua Lee, et al., Oracle database in-memory: A dual format in-memory database, in: 2015 IEEE 31st International Conference on Data Engineering, IEEE, 2015, pp. 1253–1258.

[15] Amirali Boroumand, Saugata Ghose, Geraldo F. Oliveira, Onur Mutlu, Polynesia: Enabling high-performance and energy-efficient hybrid transactional/analytical databases with hardware/software co-design, in: 2022 IEEE 38th International Conference on Data Engineering, ICDE, 2022, pp. 2997–3011.

[16] Per-Åke Larson, Adrian Birka, Eric N. Hanson, Weiyun Huang, Michal Nowakiewicz, Vassilis Papadimos, Real-time analytical processing with SQL server, Proc. VLDB Endow. 8 (12) (2015) 1740–1751.

[17] Vijayshankar Raman, Gopi Attaluri, Ronald Barber, Naresh Chainani, David Kalmuk, Vincent KulandaiSamy, Jens Leenstra, Sam Lightstone, Shaorong Liu, Guy M. Lohman, et al., DB2 with BLU acceleration: So much more than just a column store, Proc. VLDB Endow. 6 (11) (2013) 1080–1091.

[18] TPC-C Benchmark, 2010.

[19] TPC-H Benchmark, 2010.

[20] Richard Cole, Florian Funke, Leo Giakoumakis, Wey Guy, Alfons Kemper, Stefan Krompass, Harumi Kuno, Raghunath Nambiar, Thomas Neumann, Meikel Poess, et al., The mixed workload CH-benCHmark, in: Proceedings of the Fourth International Workshop on Testing Database Systems, 2011, pp. 1–6.

[21] Fábio Coelho, João Paulo, Ricardo Vilaça, José Pereira, Rui Oliveira, Htapbench: Hybrid transactional and analytical processing benchmark, in: Proceedings of the 8th ACM/SPEC on International Conference on Performance Engineering, 2017, pp. 293–304.

[22] Swarm64 HTAP Benchmark for PostgreSQL, 2021.

[23] Guoxin Kang, Lei Wang, Wanling Gao, Fei Tang, Jianfeng 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, 2022, pp. 1822–1834.

[24] Anja Bog, Kai Sachs, Hasso Plattner, Interactive performance monitoring of a composite oltp and olap workload, in: Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data, 2012, pp. 645–648.

[25] Anja Bog, Hasso Plattner, Alexander Zeier, A mixed transaction processing and operational reporting benchmark, Inf. Syst. Front. 13 (2011) 321–335.

[26] Elena Milkai, Yannis Chronis, Kevin P. Gaffney, Zhihan Guo, Jignesh M. Patel, Xiangyao Yu, How good is my HTAP system? in: Proceedings of the 2022 International Conference on Management of Data, 2022, pp. 1810–1824.

[27] Joy Arulraj, Andrew Pavlo, Prashanth Menon, Bridging the archipelago between row-stores and column-stores for hybrid workloads, in: Proceedings of the 2016 International Conference on Management of Data, 2016, pp. 583–598.

[28] Manos Athanassoulis, Kenneth S. Bøgh, Stratos Idreos, Optimal column layout for hybrid workloads, Proc. VLDB Endow. 12 (13) (2019) 2393–2407.

[29] Iraklis Psaroudakis, Florian Wolf, Norman May, Thomas Neumann, Alexander Böhm, Anastasia Ailamaki, Kai-Uwe Sattler, Scaling up mixed workloads: a battle of data freshness, flexibility, and scheduling, in: Performance Characterization and Benchmarking. Traditional to Big Data: 6th TPC Technology Conference, TPCTC 2014, Hangzhou, China, September 1–5, 2014. Revised Selected Papers 6, Springer, 2015, pp. 97–112.

[30] Aunn Raza, Periklis Chrysogelos, Angelos Christos Anadiotis, Anastasia Ailamaki, Adaptive HTAP through elastic resource scheduling, in: Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data, 2020, pp. 2043–2054.

[31] Utku Sirin, Sandhya Dwarkadas, Anastasia Ailamaki, Performance characterization of htap workloads, in: 2021 IEEE 37th International Conference on Data Engineering, ICDE, IEEE, 2021, pp. 1829–1834.

[32] Fay Chang, Jeffrey Dean, Sanjay Ghemawat, Wilson C. Hsieh, Deborah A. Wallach, Mike Burrows, Tushar Chandra, Andrew Fikes, Robert E. Gruber, Bigtable: A distributed storage system for structured data, ACM Trans. Comput. Syst. (TOCS) 26 (2) (2008) 1–26.

[33] Sergey Melnik, Andrey Gubarev, Jing Jing Long, Geoffrey Romer, Shiva Shivakumar, Matt Tolton, Theo Vassilakis, Dremel: interactive analysis of web-scale datasets, Proc. VLDB Endow. 3 (1–2) (2010) 330–339.

[34] Jeff Shute, Radek Vingralek, Bart Samwel, Ben Handy, Chad Whipkey, Eric Rollins, Mircea Oancea, Kyle Littlefield, David Menestrina, Stephan Ellner, et al., F1: A distributed SQL database that scales, 2013.

[35] James C Corbett, Jeffrey Dean, Michael Epstein, Andrew Fikes, Christopher Frost, Jeffrey John Furman, Sanjay Ghemawat, Andrey Gubarev, Christopher Heiser, Peter Hochschild, et al., Spanner: Google's globally distributed database, ACM Trans. Comput. Syst. (TOCS) 31 (3) (2013) 1–22.

[36] Ashish Gupta, Fan Yang, Jason Govig, Adam Kirsch, Kelvin Chan, Kevin Lai, Shuo Wu, Sandeep Dhoot, Abhilash Kumar, Ankur Agiwal, et al., Mesa: Geo-replicated, near real-time, scalable data warehousing, 2014.

[37] David F. Bacon, Nathan Bales, Nico Bruno, Brian F. Cooper, Adam Dickinson, Andrew Fikes, Campbell Fraser, Andrey Gubarev, Milind Joshi, Eugene Kogan, et al., Spanner: Becoming a SQL system, in: Proceedings of the 2017 ACM International Conference on Management of Data, 2017, pp. 331–343.

[38] Zhenkun Yang, Chuanhui Yang, Fusheng Han, Mingqiang Zhuang, Bing Yang, Zhifeng Yang, Xiaojun Cheng, Yuzhong Zhao, Wenhui Shi, Huafeng Xi, et al., OceanBase: a 707 million tpmc distributed relational database system, Proc. VLDB Endow. 15 (12) (2022) 3385–3397.

[39] Jianying Wang, Tongliang Li, Haoze Song, Xinjun Yang, Wenchao Zhou, Feifei Li, Baoyue Yan, Qianqian Wu, Yukun Liang, ChengJun Ying, et al., PolarDB-IMCI: A cloud-native HTAP database system at alibaba, Proc. ACM Manage. Data 1 (2) (2023) 1–25.

[40] Franz Färber, Norman May, Wolfgang Lehner, Philipp Große, Ingo Müller, Hannes Rauhe, Jonathan Dees, The SAP HANA database–An architecture overview, IEEE Data Eng. Bull. 35 (1) (2012) 28–33.

[41] Niloy Mukherjee, Shasank Chavan, Maria Colgan, Mike Gleeson, Xiaoming He, Allison Holloway, Jesse Kamp, Kartik Kulkarni, Tirthankar Lahiri, Juan Loaiza, et al., Fault-tolerant real-time analytics with distributed oracle database in-memory, in: 2016 IEEE 32nd International Conference on Data Engineering, ICDE, IEEE, 2016, pp. 1298–1309.

[42] Guoliang Li, Chao Zhang, HTAP databases: What is new and what is next, in: Proceedings of the 2022 International Conference on Management of Data, 2022, pp. 2483–2488.

[43] Mohammad Alomari, Michael Cahill, Alan Fekete, Uwe Rohm, The cost of serializability on platforms that use snapshot isolation, in: 2008 IEEE 24th International Conference on Data Engineering, IEEE, 2008, pp. 576–585.

[44] TATP Benchmark Description (Version 1.0), 2009.

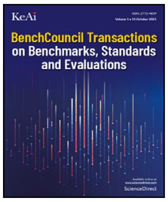
[45] Patrick E. O'Neil, Elizabeth J. O'Neil, Xuedong Chen, The star schema benchmark (SSB), Pat 200 (2007) 50.

[46] Hasso Plattner, A common database approach for OLTP and OLAP using an in-memory column database, in: Proceedings of the 2009 ACM SIGMOD International Conference on Management of Data, 2009, pp. 1–2.

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

CoviDetector: A transfer learning-based semi supervised approach to detect Covid-19 using CXR images

Deepraj Chowdhury [a], Anik Das [b], Ajoy Dey [c], Soham Banerjee [a], Muhammed Golec [d,e], Dimitrios Kollias [d], Mohit Kumar [f], Guneet Kaur [f], Rupinder Kaur [g], Rajesh Chand Arya [h], Gurleen Wander [i], Praneet Wander [j], Gurpreet Singh Wander [k], Ajith Kumar Parlikad [l], Sukhpal Singh Gill [d,*], Steve Uhlig [d]

[a] *Department of Electronics and Communication Engineering, International Institute of Information Technology, Naya Raipur, India*
[b] *Department of Computer Science Engineering, RCC Institute of Information Technology, Kolkata, India*
[c] *Department of Electronics and Telecommunication Engineering, Jadavpur University, Jadavpur, India*
[d] *School of Electronic Engineering and Computer Science, Queen Mary University of London, London, UK*
[e] *Abdullah Gül University, Kayseri, Turkey*
[f] *Department of Information Technology, National Institute of Technology, Jalandhar, Punjab, India*
[g] *Department of Science, Kings Education, London, UK*
[h] *Department of Cardiac Anaesthesia, Hero DMC Heart Institute, unit Dayanand Medical College and Hospital, Ludhiana, Punjab, India*
[i] *Chelsea and Westminster Hospital, NHS Trust London, London, UK*
[j] *St Mary's Hospital, Trinity Health of New England, Waterbury, CT, USA*
[k] *Department of Cardiology, Hero DMC Heart Institute, Dayanand Medical College and Hospital, Ludhiana, Punjab, India*
[l] *Institute for Manufacturing, Department of Engineering, University of Cambridge, Cambridge, UK*

ARTICLE INFO

Keywords:
Machine learning
Deep neural network
Transfer learning
Android app
Chest X-ray (CXR)
COVID-19
Healthcare

ABSTRACT

COVID-19 was one of the deadliest and most infectious illnesses of this century. Research has been done to decrease pandemic deaths and slow down its spread. COVID-19 detection investigations have utilised Chest X-ray (CXR) images with deep learning techniques with its sensitivity in identifying pneumonic alterations. However, CXR images are not publicly available due to users' privacy concerns, resulting in a challenge to train a highly accurate deep learning model from scratch. Therefore, we proposed **CoviDetector**, a new semi-supervised approach based on transfer learning and clustering, which displays improved performance and requires less training data. CXR images are given as input to this model, and individuals are categorised into three classes: (1) COVID-19 positive; (2) Viral pneumonia; and (3) Normal. The performance of CoviDetector has been evaluated on four different datasets, achieving over 99% accuracy on them. Additionally, we generate heatmaps utilising Grad-CAM and overlay them on the CXR images to present the highlighted areas that were deciding factors in detecting COVID-19. Finally, we developed an Android app to offer a user-friendly interface. We release the code, datasets and results' scripts of CoviDetector for reproducibility purposes; they are available at: https://github.com/dasanik2001/CoviDetector

1. Introduction

The COVID-19 virus, the first case of which is thought to have emerged in December 2019, has caused 6.9M deaths as of July 02, 2023 [1]. When infected people are coughing or sneezing, viral droplets can stay in the air for three hours. Respiratory infection harms healthy people's lungs and tissues [2]. COVID-19 variant instances have increased rapidly in recent months [3]. However, some international researchers think that this increase will decrease by 2024, and the world may become normal [4]. At present, COVID-19 detection is being performed using one of the below three tests:

[*] Correspondence to: School of Electronic Engineering and Computer Science, Queen Mary University of London, London, E1 4NS, UK.
E-mail addresses: deepraj19101@iiitnr.edu.in (D. Chowdhury), anikdas.2113@gmail.com (A. Das), deyajoy80@gmail.com (A. Dey), soham20101@iiitnr.edu.in (S. Banerjee), m.golec@qmul.ac.uk (M. Golec), d.kollias@qmul.ac.uk (D. Kollias), kumarmohit@nitj.ac.in (M. Kumar), guneetkw.it.22@nitj.ac.in (G. Kaur), rupinderchem@gmail.com (R. Kaur), drrajesharya@yahoo.com (R.C. Arya), gurleenwander@gmail.com (G. Wander), praneet_wander@hotmail.com (P. Wander), drgswander@yahoo.com (G.S. Wander), aknp2@cam.ac.uk (A.K. Parlikad), s.s.gill@qmul.ac.uk (S.S. Gill), steve.uhlig@qmul.ac.uk (S. Uhlig).

https://doi.org/10.1016/j.tbench.2023.100119
Received 6 June 2023; Received in revised form 3 July 2023; Accepted 3 July 2023
Available online 6 July 2023

- Computed Tomography (CT) scans of chest that use three-dimensional radiographs and are a key diagnostic tool. However, not every health care institution has the facility of CT scan with them.
- Ribonucleic Acid (RNA), which can be detected from nasopharyngeal swabs using the Reverse Transcription Polymerase Chain Reaction (RT-PCR) technique. However, this technology is hard to get to, and is time consuming.
- Chest X-ray (CXR); the necessary equipment for a CXR is readily available and is more transferable and quick compared to CT-scan ones. CXR examinations are also fast as they require only roughly 15 s for each participant.

As COVID-19 continues to sweep the globe, researchers and data scientists have begun employing deep learning methods for the automated identification of the virus in humans [5]. Artificial Intelligence (AI)-based computer-aided diagnostics has advanced rapidly in several domains of medicine because of contemporary advances in the field of AI [6]. Diseases like cancer can be diagnosed with greater accuracy thanks to automatic image analysis performed via deep learning and in more detail Convolutional Neural Networks (CNN) [2]. Therefore, these models show promise for enhancing the use of CXR images in the diagnosis of COVID-19 [7].

AI systems have previously been utilised to correctly identify pneumonia from CXR [8]. Differentiating between viral and bacterial pneumonia has been performed via the use of deep learning. K-Nearest Neighbour (KNN), Support Vector Machine (SVM), and CNNs are only a few of the AI classification strategies used [9]. Among the machine learning algorithms for classification tasks, KNN is considered one of the easiest [10]. The KNN algorithm basically considers the similarity distance between the new and already available data and classifies the new data accordingly [11]. SVM is another classification algorithm, that primarily creates the best possible boundary that can separate n-dimensional space into classes with the aim to classify the new data into one of the classes [12]. CNN is effective in image classification to recognise COVID-19 [13–15]. Multi-layer neural networks, like CNNs, are the key to the system's success in recognising visual trends. Various pre-trained CNN models, including AlexNet, VGG16, InceptionV3, and DenseNet are available, among which InceptionV3 demonstrates better accuracy and performance for the COVID-19 classification [16].

1.1. Motivation

CXR-based identification of COVID-19 patients might be hampered by a lack of effective and experienced medical experts, especially in rural areas [17]. So, there is a requirement for a simple and inexpensive deep learning-based technique to identify COVID-19 patients within a short span of time [18]. This model will be available to all patients, even though doctors may not be available [19].

CXR of COVID-19-affected lungs show less porosity or visibility because the lungs are stuffed with smooth and dense mucus [20]. While multiple algorithms and diagnostic tools based on machine and deep learning have shown promise, they still fall short of high performance in terms of precision and error rate [21]. Therefore, healthcare professionals and the community as a whole would benefit from choosing a group of effective deep learning-based analysts.

In this paper, CoviDetector is a machine learning system that utilises transfer learning and a deep learning model (InceptionV3) for the early identification and diagnosis of COVID-19 by chest X-rays. Moreover, this system is deployed as an Android Backend. The android application user interface (UI) is designed using the React Native framework, which takes as input of an image and predicts and displays the results on the screen.

CoviDetector aims to provide an Android application for the user, which will allow the user to predict COVID-19 automatically using CXR images. In order to have quick, accurate, economical, and hassle-free COVID-19 recognition, we compare the performance of various deep learning models across various metrics, including accuracy, precision, recall, F-score, and sensitivity, with the ultimate goal of applying the most effective model on the back-end of an Android app.

1.2. Problem statement

Detection of COVID-19 can be done with a RT-PCR test, which can only be conducted in a laboratory environment; tests are performed by taking a swab from the nose. The time required for this test's results varies between 8 and 24 h. Therefore, alternative methods, such as rapid antigen testing, have emerged. However rapid antigen tests are not accepted by many medical practitioners because of their high false positive and false negative rates. For this reason, a COVID-19 detection method that takes as input CXR images is used; this method is both fast and reliable. In this paper, we propose a smartphone application that detects COVID-19 using deep learning (DL) and CXR images (as an alternative to RT-PCR tests). Users can quickly learn if they have COVID-19 by uploading CXR images to the Android app. We apply several DL algorithms to CXR images to detect COVID-19. These models are trained with unbiased and balanced data, so the prediction results are unbiased and accurate. Experimental results have shown that the models display high performance in detecting COVID-19.

1.3. Our contributions

The main contributions of this work are:

- an Android application compatible with smartphones that diagnoses COVID-19 through CXR images. The diagnosis is performed from CXRs via the proposed semi-supervised method that consists of a Deep Neural Network (DNN) and K-means clustering; the proposed methodology also utilises transfer learning. Finally, GradCAM is used to highlight the areas in the CXRs that were the deciding factors in the method's decision in detecting COVID-19;
- the DNN model trained with InceptionV3 CNN blocks is the best performing model for detecting COVID-19.

CoviDetector is an Android app in development with the intention of providing medical professionals and consumers with an easy way to check for Covid-19. The proposed App helps patients receive proper classification results for the Chest X-rays uploaded by them. This would primarily help the doctors take immediate action without waiting for reports and start instant treatment. CoviDetector can accurately determine if a person is COVID-19 positive or negative by analysing only an image of CXR. This information, as we have previously mentioned, can be used by doctors to make instant decisions. This would also help society, as the resources utilised are just an Internet connection with no manpower or industrial interference.

The rest of the paper is organised as follows: Section 2 presents related work. Section 3 introduces a proposed methodology, and Section 4 presents the experimental setup and results. Finally, Section 5 concludes the paper and highlights future directions.

2. Related work

Over the past several months, a growing body of research has evaluated the efficacy of deep-learning models for the identification of COVID-19 in CXR images. This section includes short discussions of a few of these works that are relevant to our own.

Accuracy of 98.93%, specificity of 98.66%, precision of 96.39%, and F-1 score of 98.15% were achieved with the approach described by Mittal et al. [22]. There were a total of 1215 images in their collection, including 250 from COVID-19. COVIDiagnosis-Net was developed by Ucar et al. [23], and the precision across all three classes was 98.26%. The DNN model presented by Ahammed et al. [24] has achieved 94.03 percent accuracy using the CNN. The researchers have used data from

Table 1
Comparative study of relate works for COVID-19 detection.

Works	Dataset used	Methodology	Accuracy(%)	Android app	Transfer learning	Training data Size
Xinggang et al. [28]	Custom Gathered Dataset	UNet based 3D DNN	90.10%	✗	✓	229 No-Findings & 313 Covid-19
Yujin et al. [29]	CoronaHack & Other Datasets	ResNet18 based FC-DenseNet	91.90%	✗	✓	134 No-Findings & 126 Covid-19 & 94 Others
Ahmed et al. [30]	IEEE CovidCXR Dataset	CNN based Approach	94.00%	✓	✗	1341 No-Findings & 1200 Covid-19
Ozturk et al. [26]	CohenJP Dataset	DarkNet-19 based CNN	98.08%	✗	✗	500 No-Findings & 125 Covid-19 & 500 Pneumonic
Ucar et al. [23]	CovidX Dataset	Deep Bayes-SqueezeNet based Approach	98.26%	✗	✗	1229 No-Findings & 1229 Covid-19 & 1229 Others
Bushra et al. [31]	CohenJP & other datasets	Tensorflow Lite based CNN	98.65%	✓	✗	592 No-Findings & 592 Covid-19
Taresh et al. [32]	COVID-19 Radiography Database	VGG16 based CNN	98.72%	✗	✓	1140 No-Findings & 820 Covid-19 & 1150 Pneumonic
Ahsan et al. [33]	CohenJP & CovidCXR Datasets	Feature Fusion based CNN	99.49%	✗	✗	2489 No-Findings & 1584 Covid-19
CoviDetector (This Paper)	**CovidCXR, NIH CXR, DLAI3 datasets**	**InceptionV3 based CNN and Clustering**	**99.69%**	✓	✓	4253 No-Findings & 3160 Covid-19 & 6034 Others

three categories to train the algorithm. In this case, too, the dataset had a rather low sample size, which is not optimal for trying to train a deep learning-based system for COVID-19 diagnosis.

The ResNet101 CNN model was also employed by Azemin et al. [25]. The result of their work was a thousand images that were fed into the pre-trained model. They were just 71.9% accurate at best. Ozturk et al. [26] combined DarkCovidNet with a collection of 1125 photos, 125 of which were taken from COVID-19 examples, to develop a framework. An overall accuracy of 98.08 percent was found in a 5-fold cross-validation of binary tags. Utilising algorithms like ResNet50, VGG16, VGG19, and DensNet121, Khan et al. [27] constructed a novel framework for diagnosis of CXR images; VGG16 and VGG19 demonstrated higher accuracies of approximately 99.3 percent in contrast to others.

Yujin et al. [29] employed a patch-based CNN technique for a much lesser amount of trainable parameters for COVID-19 diagnosis, which they attributed to their use of a segmented network-based approach. When taking into account the increased sensitivity of their line of work, the 91.9% correctness they attained is rather impressive. In a related paper, Fan et al. [34] developed Inf-Net with a parallel partial decoder to combine characteristics at a higher level. Their method was 97.4 percent accurate while also being 87.1 percent sensitive. To forecast COVID-19 CT scans, Xinggang et al. [28] also presented a 3D deep neural network. The precision of their work was 90.1%. One CT volume from a single patient was processed by the algorithm in just 1.93 s, making it one of the quickest models ever created.

A further investigation aimed to identify COVID-19 using a transfer learning strategy and three pertained models was conducted by Loey et al. [35]. Correctness for all three categories on AlexNet was 85.20 percent using a dataset of over 300 X-ray pictures that included around 70 photos of COVID-19. In order to enhance the ResNet-101 and ResNet-151's weights during training, fusion effects were used, and Wang et al. [36] were able to increase the model's accuracy to 96.1%. Mahmud et al. [37] also obtained a success rate of 97.4 percent for binary classes using a CNN model they devised called CovXNet. A deep learning method was described by Minaee et al. [38] to identify COVID-19 from CXR. Using data augmentation, they generated modified pictures of the CXR plates, and their approach had a sensitivity of 98% and a specificity of 90%.

In another study on COVID-19 detection, Chakrabarti et al. [32] employed ensemble learning in conjunction with a Deep Convolutional Neural Networks (DCNN) to predict binary classes. They employed an aggregate of 1006 COVID-19 suspected patient's pictures (538 positive and 468 negative) to evaluate the performance of the model. The degree of precision they achieved was 91.62 percent. Ahmed et al. [30] also used Tflite to develop a method for developing mobile applications that relied on CNNs. They found that their method, on average, was 94% accurate. Ahsan et al. [33] found similar success with feature fusion and deep learning. The recommended approach improved accuracy to 99.49 percent, performing better than any single CNN.

To boost the overall performance of COVID-19 methods of classification, Tabik et al. [39] used a Smart Data Based network called COVID-SDNet. Their method had a 97.72 percent success rate. Taresh et al. [40] utilised transfer learning to identify COVID-19 from CXR images, so it is possible to do the same. With a 98.72 percent accuracy rate, their VGG16-based model was superior to the competition.

2.1. Critical analysis

Table 1 compares **CoviDetector** with existing transfer learning and DNN models. The accuracy of the aforementioned study works is pretty high, yet it is also important to put it into practise in a way that could be accessible to patients in general. Only two of the reported studies have even investigated using transfer learning in a user interface for an application supported by the model. A novel framework that enhances accuracy and implementation in an Android App is required as very little work has been done to construct a CNN-based Android App, specifically for COVID-19 diagnosis. The required framework needs to be proposed in order to (1) facilitate communication between users, (2) incorporate the most effective method with better precision, and (3) guarantee the highest level of care for both patients and medical professionals. To fill these knowledge shortcomings, we propose a novel system called CoviDetector to improve research and address the above-mentioned challenges.

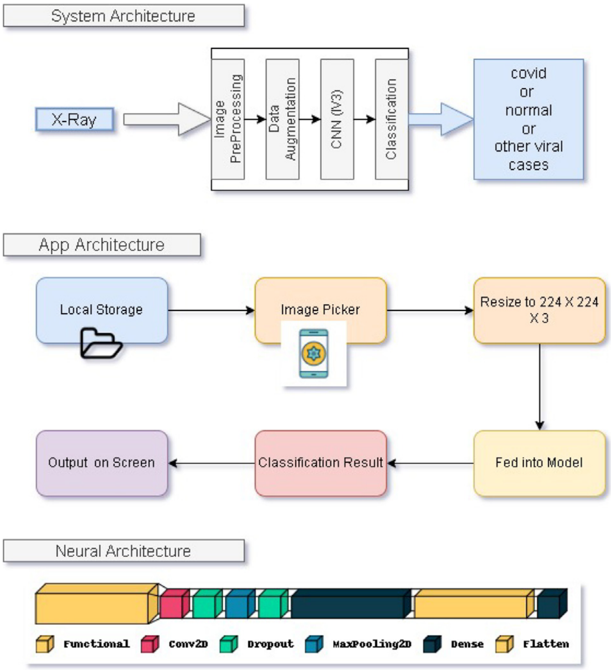


Fig. 1. CoviDetector Architecture: The first portion of the figure depicts the layers in the System Architecture i.e the basic internal working of the CoviDetector application. The next portion is the App Architecture which shows the functioning of the Application UI to fetch the inputs from the user and how it is fed to the model for evaluation. The next and last portion of the image picturises the sequence of layers in the Neural Architecture used in the CoviDetector model.

3. CoviDetector: Proposed methodology

Methods for accomplishing the goals of the proposed work are discussed in detail below, with specific attention paid to the system architecture that was developed to accomplish these goals including input preprocessing of the image inputs followed by labelling appropriate classes of data, and datasets used, continuing with the extraction of features, to CNN-based classification.

3.1. System architecture

CXR images are used as input for the proposed technique for COVID-19 detection. To begin, this system shifts user-provided photos into the more widely used Red–Green–Blue (RGB) colour space. Additionally, the algorithm only takes into account photos that are comparable to CXR. To begin, a Structural Similarity Index Measure (SSIM) is applied to the picture in order to determine its structural similarity with a CXR. If that happens, just that picture will be considered in further analyses. For example, if the image has a SSIM value greater than the threshold value (this value depends on the type of application) with the image of CXR, then that image will be considered for prediction; otherwise, that image will be discarded. The InceptionV3 model is quite effective in obtaining features and picture classification. The whole model was implemented using an Android Front-end made using the React-Native framework and TensorflowJS as a backend. The System Architecture has been visualised in Fig. 1. The system takes CXR images as input, classifies them into different classes, and gives the predicted class as the output.

3.2. Input pre-processing

Image pre-processing is a vital step to achieving meaningful and accurate classification. Therefore, there is a need to resize all images to 224 × 224 × 3 pixel resolution and their intensity values to be normalised to $[-1, 1]$ (by dividing with 255). Subsequently an 80%–20% ratio between training and testing sets have been applied to every dataset.

3.3. Data augmentation

The dataset included a highly imbalanced number of samples from various classes. We first used the data augmentation technique to increase the number of sample images for every class in order to address the class imbalance. Random cropping and horizontal & vertical flips were applied for the augmentation of existing data. In order to equalise the number of samples from each class, the following stage included selecting the class with the fewest images and extracting random samples from other classes. The researchers were able to create a more optimal model using the larger sample size. The size of the training dataset has been increased by data augmentation. Further, random cropping, and horizontal & vertical flips have been applied to improve the robustness of the training model.

3.4. Model

This section gives an overview about deep learning models used in this research work.

3.4.1. VGG16

The VGG16 network is a CNN model with 1000 outcomes. It has 13 convolution neural layers and three layers that are fully connected. It is capable of handling 224 × 224 pixel colour pictures. After that, many convolution networks are used to determine if the layer is red, green, or blue. Both the input and the resultant feature maps have the same size in this scenario. The field of reception of any convolution filter is just 3 × 3 in stride 1. Row and column padding are used following convolution to preserve spatial resolution. There are 13 convolution layers and 5 max pool layers, as has already been stated. The largest pooling window is 2 strides by 2 strides. VGG16's architecture and primary feature, transfer learning, are both formed by [41]. For the purpose of using CNNs as a fixed feature extractor1, a CNN architecture trained on a large dataset is taken, and its final fully connected layer is removed. For this new dataset, the remainder of the CNN serves as a fixed-feature extractor. Let us assume that there is a model that is trained on the basics of one set of databases, like ImageNet and an application that is trained on the basics of some other database, like Pascal. When an image enters the first layer, consider only the edge. Then it moves to the second layer, which considers corners, curves, etc. On further moving to the third layer, it considers the features of the high-level layer. On further moving more deeper, the domain becomes more specific.

3.4.2. VGG19

To put it simply, VGG19 is a state-of-the-art convolutional neural network. The visual geometry group at Oxford has put forth this idea. Including its 16 convolutional layers, 3 fully connected layers, 5 max pool layers, and 1 softmax layer, the VGG19 model is rather complex. This phenomenon has been designated as VGG19. It has been taught with millions of different photos, giving it a great depth of understanding. Colour photos of a certain width and height are acceptable. After that, the pictures go through a series of convolution networks, one for each colour channel. Both the input and the resultant feature maps have the same size in this situation. The receptive field size of every convolution filter is exactly 3 × 3 stride 1. Use row and column padding after convolution to keep spatial resolution stable. The architecture of the network consists of 13 convolution layers and 5 max pool layers, as stated before. The largest allowable pooling window size is 2 by 2 steps. VGG19 borrows its model architecture on like its predecessor, VGG16. When pitted against VGG16, VGG19 performs somewhat better. It represents an idea in terms of form, colour, and architecture. If the picture is in the ImageNet database, a pre-trained version of the network can be loaded and used. After being taught, the network can sort photos into one of a thousand distinct categories, each of which can include commonplace items like a computer keyboard, mouse, or pencil.

3.4.3. DenseNet121

The DenseNet121 model is part of the larger DenseNet family of image classifiers. The DenseNet121 model differs mostly in terms of its larger size and greater precision. The DenseNet121 model is considerably bigger than its smaller counterpart. They were originally developed using Torch, but the authors have since ported them to Caffe. Pre-training on ImageNet has been done for the DenseNet models. DenseNet121's model results are representative of those of an object classifier applied to a dataset of 1000 classes from the ImageNet database. A single picture with the coordinates (1, 3, 224) in BGR order is the input to the simulation. Fewer interconnections between layers near the input and the ones near the output allow convolutional networks to be significantly deeper, more precise, and simpler to train, as proven in recent research [42]. The article employs a feed-forward neural network architecture called a Dense Convolutional Network (DenseNet). Layer data for DenseNet121 has been retrieved from [42]. Every level feeds its own feature maps into the layers above it, and each layer above it feeds its own feature maps into the levels below it. Using DenseNets has been shown to drastically cut down on the number of parameters. DenseNets achieves great performance with less memory and compute while significantly outperforming the latest developments on the majority of them.

3.4.4. InceptionV3

The InceptionV3 transfer learning method using weights from publicly available ImageNet data will serve as the focus of this research. There are 230 "frozen" layers in the model, representing parameters that should not be modified throughout the training process. Developing a model from preexisting models was shown to be more efficient than developing a new deep learning model from beginning [43]. Transfer learning allows us to retrain the final layer of an existing model, resulting in a significant decrease in not only training time, but also the size of the dataset required. One of the most known models that can be used for transfer learning is Inception V3. This model was originally trained on over a million images from 1000 classes on some very powerful machines, which resulted in highly accurate classification. Inception V3 mainly centres on consuming less computational power by modifying the previous Inception architectures. Compared to VGGNet, Inception Networks (GoogLeNet/Inception v1) have proven to be more computationally practical, both in terms of the number of parameters generated by the network and the memory and other resources used.

3.4.5. EfficientNet

EfficientNet, which was first introduced in Tan and Le's 2019 paper [44], is one of the best algorithms for typical image categorisation transfer learning tasks and ImageNet, where it has achieved State-of-the-Art efficiency. In terms of model size, the lowest base model is competitive with MnasNet, which obtained near-SOTA with a much smaller model. EfficientNet introduces a heuristic approach to model scaling, producing a set of models (B0–B7) that strike a good balance between quickness and precision when the scale is increased or decreased. However, numerous factors limit the resolution, depth, and width options. Resolutions that are not divisible by eight, sixteen, or twenty-four result in zero-padding along the end points of some layers, wasting computational resources. This is particularly true for the model's smaller variations, which is why the input resolutions for B0 and B1 are set to 224 and 240, respectively. EfficientNet's construction blocks require channel sizes to be multiples of eight. When depth and width can still be increased, memory constraints may stifle resolution. In this case, increasing depth or width while maintaining resolution may still increase performance.

3.4.6. K-means clustering

Clustering is a popular interactive data analysis method for quickly understanding how the data is organised. Data clustering is the process of identifying groupings within a dataset where individual data points have many similarities but those corresponding to different clusters have few. Using iterative steps, the K-means approach seeks to divide the dataset into K distinct, non-overlapping subgroups (clusters), with each cluster containing a single value. It makes an effort to keep clusters as dissimilar (far) as practicable while maintaining intra-cluster data points as closely related as reasonable. The algorithm groups data points into clusters with the goal of minimising the sum of squared distances among them and the centroid of the cluster (the mathematical average of all data points in that group). The homogeneity (similarity) of data points within a cluster increases as inter-cluster variation decreases.

3.5. Algorithm

In this research, we use transfer learning to train a CNN Model. The weights that are shared between model layers serve as a means of information transfer. A Convolutional 2D layer with ReLu activation and a Dropout layer follow this. This brings the total number of layers to 5. A MaxPooling Layer comes next, followed by a Flatten Layer for communication. The necessary number of classes with output is then sent to a softmax-activated dense layer that serves as the ultimate output layer. The following phase was to assemble the model with two primary variables called optimiser and loss. It has become common practise to use Binary CrossEntropy as the loss function for binary classification jobs, and Categorical CrossEntropy for multiclass data. It turned out that a learning rate of 0.0001 yielded the best outcomes from the RMSprop optimiser. The algorithm for the configuration of the model is visualised in Algorithm 1.

Algorithm 1 Model Input and Architecture of CoviDetector:

```
Require: $a$ : $data$, $b$ : labels, $z$ : number of images, $m, n$ : image dimensions,
    $f$ : Base Model, $g$ : Head Model
    $f.out$ : Output Layers of Functional Model
for i=0 to z-1 do
    $b \Leftarrow image$
    $image \Leftarrow image_{cvtcolor}$
    $image \Leftarrow image_{resige}(m, n)$
    $a \Leftarrow image$
    i++
end for

    Model(a):
    $f \Leftarrow$ VGG16,VGG19;DenseNet121;
    InceptionV3; EfficientNetB4
    $g \Leftarrow$ f.out(output layer of f)
    $g \Leftarrow$ Conv2D(g)
    $g \Leftarrow$ Dropout(g)
    $g \Leftarrow$ MaxPool2D(g)
    $g \Leftarrow$ Flatten(g)
    $g \Leftarrow$ Dense(g)
    return metric
```

3.6. The prototype application

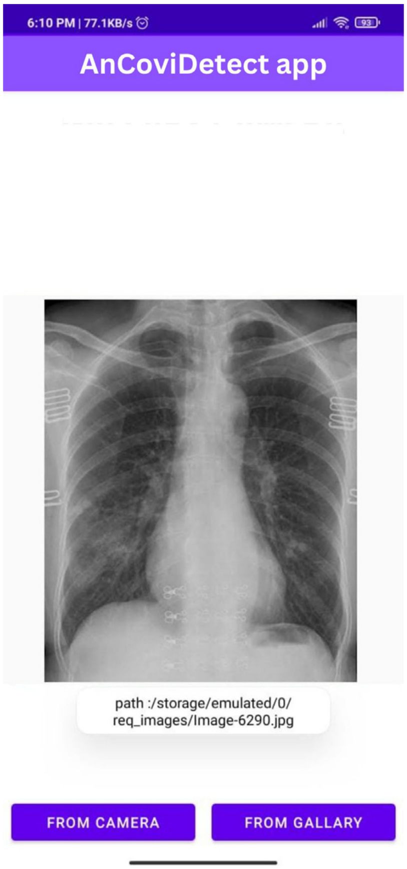
In line with the proposed model, an Android-based mobile app has been created to distinguish between Covid-19 positive and negative patients. Because of this, anyone may access a CXR picture on their computer and input it into the programme. The image is then assessed by the programme using the provided model, and a classification label is returned. The user interface prototype is shown in Fig. 2 .

Table 2
Comparative view of data splitting for multiple datasets:.

Dataset	Training (70%)	Testing (20%)	Validation (10%)	Used samples
COVID-19 CXRImage Dataset (Research)	1125	321	160	1608 out of 1823
DLAI3 Hackathon Phase3COVID-19 CXR Challenge	762	218	110	1089 out of 5507
COVID-19 RadiographyDatabase	7140	2040	1020	10.2k out of 42.3k
Covid19 Detection	7560	2160	1080	10.8k out of 24.8k

Table 3
Performance metrics 1 (Dataset1).

Model	MCC	Sensitivity	Specificity	AUC score	Training time (sec/epoch)
VGG16	0.9667	0.9717	1.0000	0.9858	31
VGG19	0.9668	0.9811	1.0000	0.9905	36
DenseNet121	0.9853	1.0000	0.9917	0.9948	30
InceptionV3	0.9876	1.0000	1.0000	0.9959	24
EfficientNetB4	0.7635	0.8361	0.8695	0.8571	40
Semi-supervised	0.9916	0.9958	0.9962	0.9937	20



path :/storage/emulated/0/
req_images/Image-6290.jpg

Fig. 2. The Prototype UI.

3.7. GradCAM

GradCAM is a kind of post-hoc attention. The term post-hoc attention means it is a method used for heatmap generation that is subsequently applied to a pre-trained neural network after training is complete and weights are known. GradCAM is a generalisation of CAM (Class Activation Mapping), and it can be applied to any CNN architecture. The basic idea behind the usage of GradCAM over here is to exploit the spatial information preserved using convolutional layers, in order to comprehend which parts of the input image played a pivotal role in the classification decision. It uses a feature map produced by the last convolutional layer of a CNN architecture (like CAM). We have applied GradCAM visualisation to DenseNet21, VGG16, and InceptionV3 algorithms, which were among the top-performing models for the datasets used. A few more reasons to use GradCAM are: (1) It does not change the architecture of the model and just gets added to it, and (2) It is class-discriminative using localisation techniques.

4. Performance evaluation

We have evaluated four different DNN models on the dataset and then analysed the accuracies obtained using the discussed approach. We examined the model's efficiency using a variety of loss functions and parameter settings before settling on a good one. As a last step, we integrated the Tensorflow backend into an Android app; the details are presented next.

4.1. Experimental setup

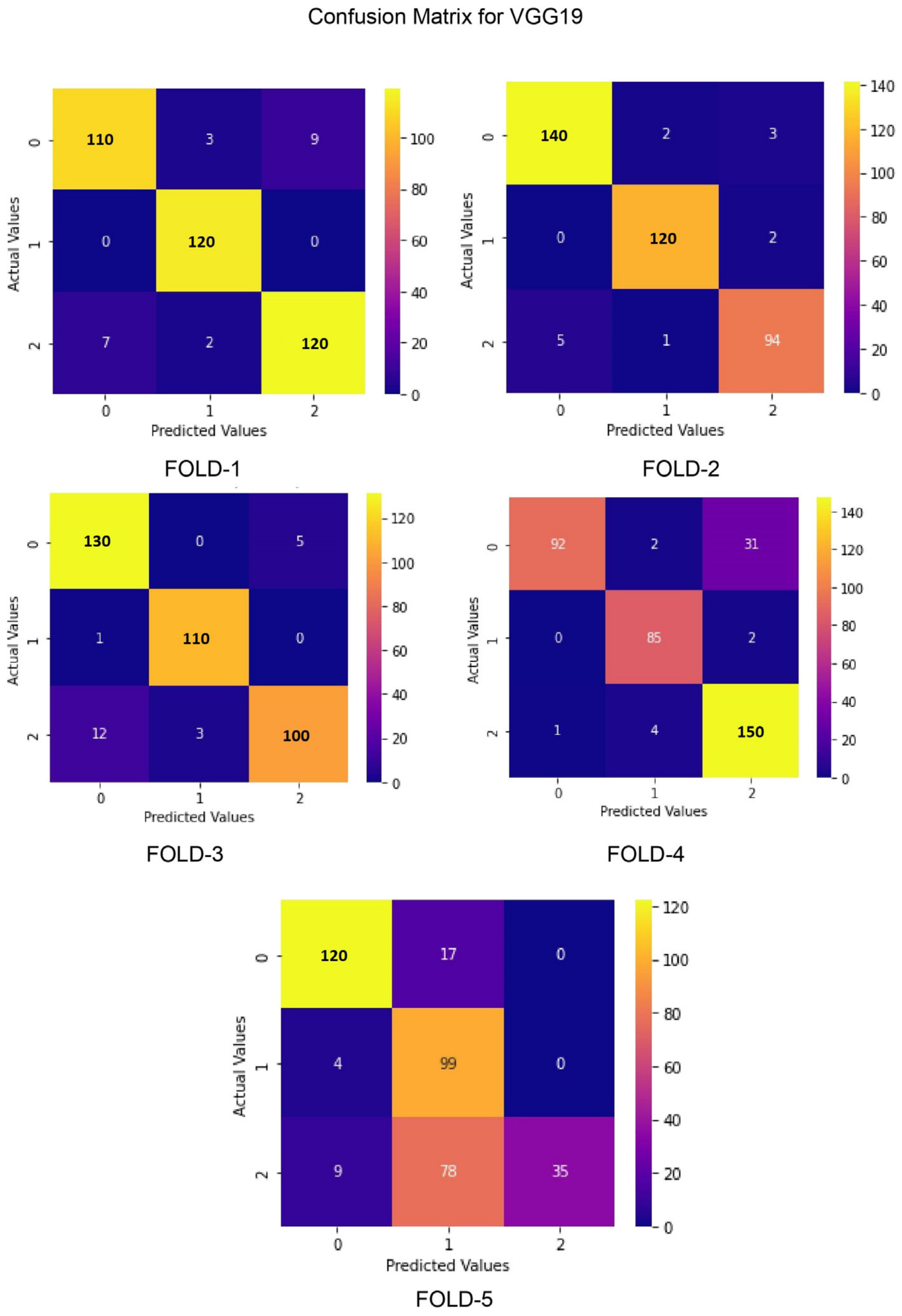
Accuracy, sensitivity, and specificity were taken into account for analysis in order to evaluate the effectiveness of several models and obtain the most effective outcomes. After training for around 20 epochs using the RMSProp optimiser and a Learning Rate of 0.0001, all of the CNN models were ready for testing. Model training takes between 31 and 35 s per epoch on VGG16 and VGG19, respectively. In InceptionV3, the time required for each epoch was around 24 s, but in DenseNet, the time required was approximately 30 s. EfficientNetB4 took about 40 s per epoch for the same data. Table 2 gives the insight of training testing and validation ratio of the datasets used. Table 3 summarises the average training time of the models.

4.2. Configuration settings

We developed the model using Tensorflow 2.2.0 and Python 3.6. NVIDIA Tesla K80 GPU has been used for the training procedure.

4.3. Dataset used

We used various datasets for experimental purposes. The authors of the datasets have accomplished the hectic task of gathering and categorising the CXR Images. There are four datasets on which experiments were performed. Two of them contain data from three classes. The other two datasets consist of data from 4 and 5 different classes. The first dataset, COVID-19 CXR Image Dataset (Research) [45] named as Dataset1 contains classes named COVID, Normal, and Viral. This dataset was used for initial model testing and validation. However, various other datasets were utilised to check the competency of the model. Another other 3 class dataset, DLAI3 Hackathon Phase3 COVID-19 CXR Challenge [46] named Dataset2 contains COVID, No-FINDING, ThoraxDisease. This dataset was also used for the validation and testing of the DNN models. The other dataset, COVID-19 Radiography



FOLD-1

FOLD-2

FOLD-3

FOLD-4

FOLD-5

Fig. 3. Confusion Matrix:VGG19 on 5-Folds(0-Normal, 1-COVID19,2-Viral).

Database [47] named Dataset3 contains 4 classes of data namely COVID, Lung Opacity, No-Finding, and Viral Pneumonia. The best model was trained with these four dataset data points where two classes were merged to form a single class of data. This model was further evaluated on the training data and on a 5 class dataset, Covid19 Detection [48] named Dataset4 which consists of data from COVID, Fibrosis, Normal, Viral and Pneumonia, to further prove the competency of the work. Apart from that, all the datasets were split into training and testing sets with an 80:20% ratio and used for 5-fold cross-validation. The performances of four different models applied to these datasets are shown in Figs. 3–6. As can be seen in Figs. 3–7 the accuracy rate of the model decreases as the number of fold operations increases. For example, the accuracy rates for Figs. 3–5 fold-1 process are 94%, 97.29% and 95.06%, respectively. For the same shapes, these rates decrease to 91.71%, 94.02% and 90.95%, respectively, after the fold-5 process.

4.4. Analysis of results

The findings of the experiments are analysed and discussed below:

4.4.1. Results on 3 class dataset

As soon as it comes to COVID-19 detection, a True Positive (TP) happens if both the patient's other investigations and the model agree that COVID-19 is present, whereas a True Negative (TN) occurs when both the patient's other investigations and the model agree that COVID-19 is not present. If a person does not have COVID-19 but the model predicts positive, we say that person has a False Positive (FP), whereas if the person possesses COVID-19, we say that the model gets a False Negative (FN).

$$Accuracy = \frac{TP + TN}{TN + TP + FN + FP} \tag{1}$$

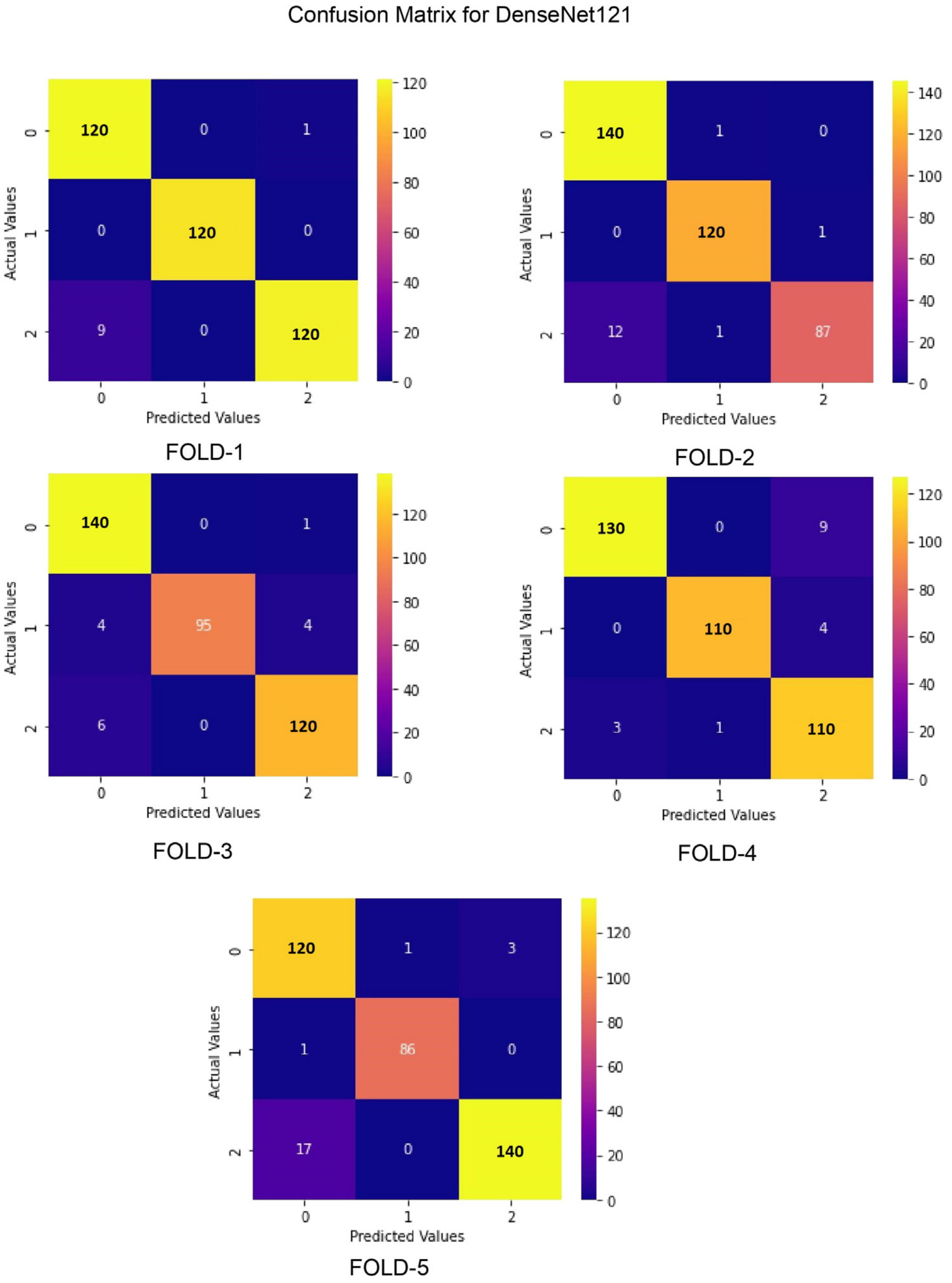


Fig. 4. Confusion Matrix:DenseNet121 on 5-Folds(0-Normal, 1-COVID19,2-Viral).

$$Sensitivity = \frac{TP}{TP + FN} \tag{2}$$

$$Specificity = \frac{TN}{TN + FP} \tag{3}$$

$$Precision = \frac{TP}{TP + FP} \tag{4}$$

$$F - Score = \frac{TP}{TP + (0.5)(FN + FP)} \tag{5}$$

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

$$MCC = \frac{TP * TN - FP * FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{7}$$

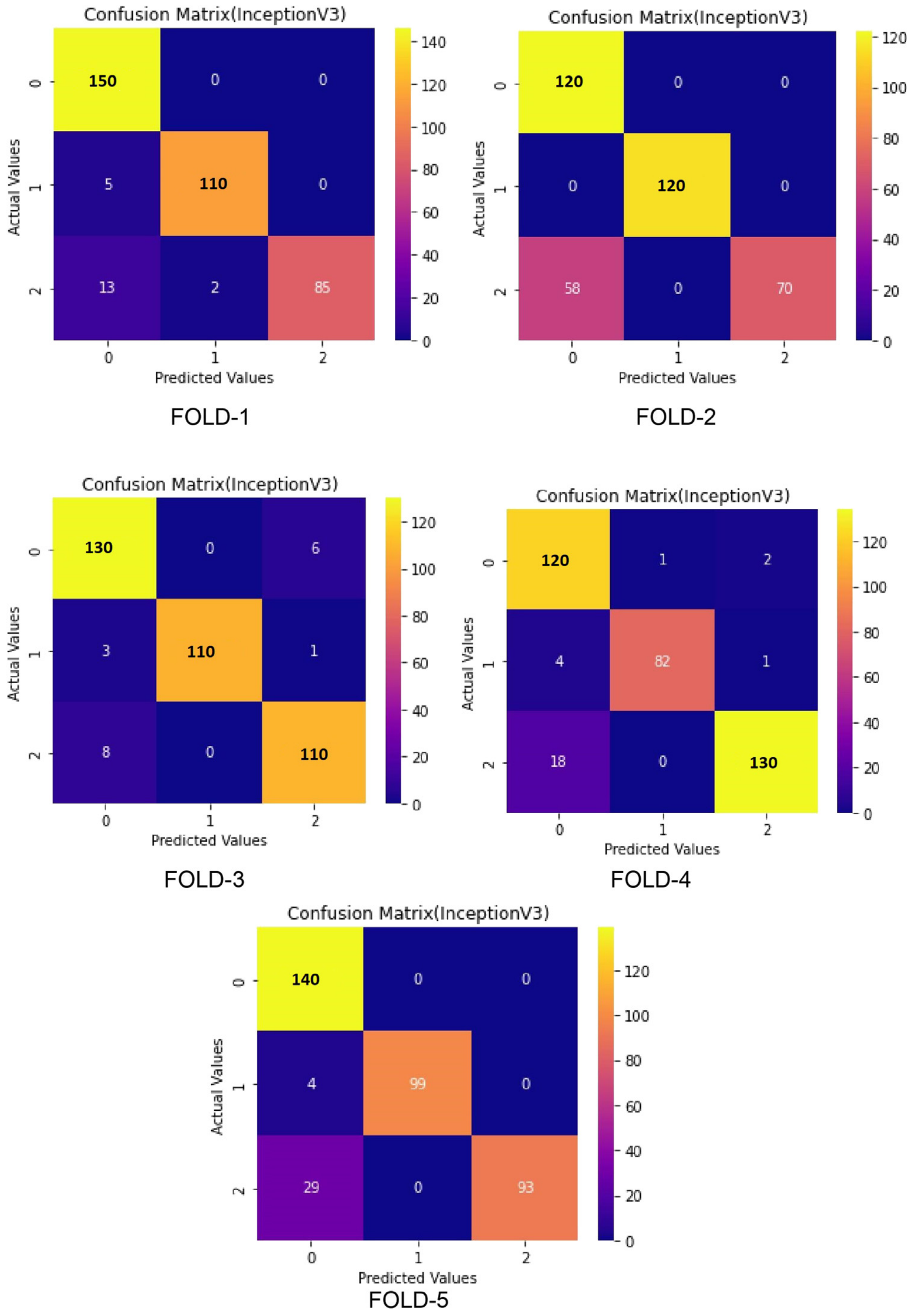
The data visualisations show a consistent trend, with training and testing accuracy improving and loss decreasing as the number of epochs grows.

Table 4
Performance metrics 2 (Dataset1).

Model	Accuracy	Precision	Recall	F-Score
VGG16	0.9876	1.0000	1.0000	1.0000
VGG19	0.9917	1.0000	1.0000	1.0000
DenseNet121	0.9958	1.0000	0.9917	0.9958
InceptionV3	0.9965	1.0000	1.0000	1.0000
EfficientNetB4	0.7863	0.8026	0.7685	0.7553
Semi-supervised	0.9969	0.9989	1.000	0.9971

Table 3 shows the MCC (Matthews correlation coefficient), sensitivity, specificity, and AUC Score values for each model (see 4.4.3 for Semi-supervised model). Whereas accuracy, precision, recall and F1 score are also tabulated in Table 4.

Fig. 8 not only depicts the confusion matrix but additionally the AUC-ROC, or true positive rate (TPR), vs. false positive rate (FPR), curve for the actual models used. Other Important Metrics include the



FOLD-1

FOLD-2

FOLD-3

FOLD-4

FOLD-5

Fig. 5. Confusion Matrix:InceptionV3 on 5-Folds(0-Normal, 1-COVID19,2-Viral).

Accuracy versus Epoch Curve and the Loss versus Epoch curve, which are visualised in Fig. 9. From Fig. 8, it is analysed that the proposed algorithm has a very high true positive rate in comparison to a very high false positive rate, which signifies that the predicted results are mostly correct for a positive response, whereas Fig. 9 shows the loss and accuracy the model is having after each epoch to get an optimal epoch count.

4.4.2. Comparison of different datasets

The results of training and testing various transfer learning models on three different datasets are visualised in Fig. 10. To further evaluate the CoviDetector model, the best-performing model, InceptionV3 was evaluated on a small dataset after being trained on a separate dataset. Moreover, the InceptionV3 model was trained on one Dataset in which 4 classes were converted into 3 and then the model was further tested on two other datasets containing 5 classes of data that was converted into 3 classes. Both the results are visualised in Figs. 12–14.

4.4.3. Semi-supervised ML

Apart from deep learning and transfer learning models, a semi-supervised method of machine learning was also implemented for image clustering [49–51]. In this paper, the K-means clustering model was implemented as a semi-supervised learning algorithm. The K-means clustering technique works on a specified number of clusters; in this scenario, the clusters varied from 2 to 20 different clusters. This method of getting the optimal number of clusters is known as the elbow

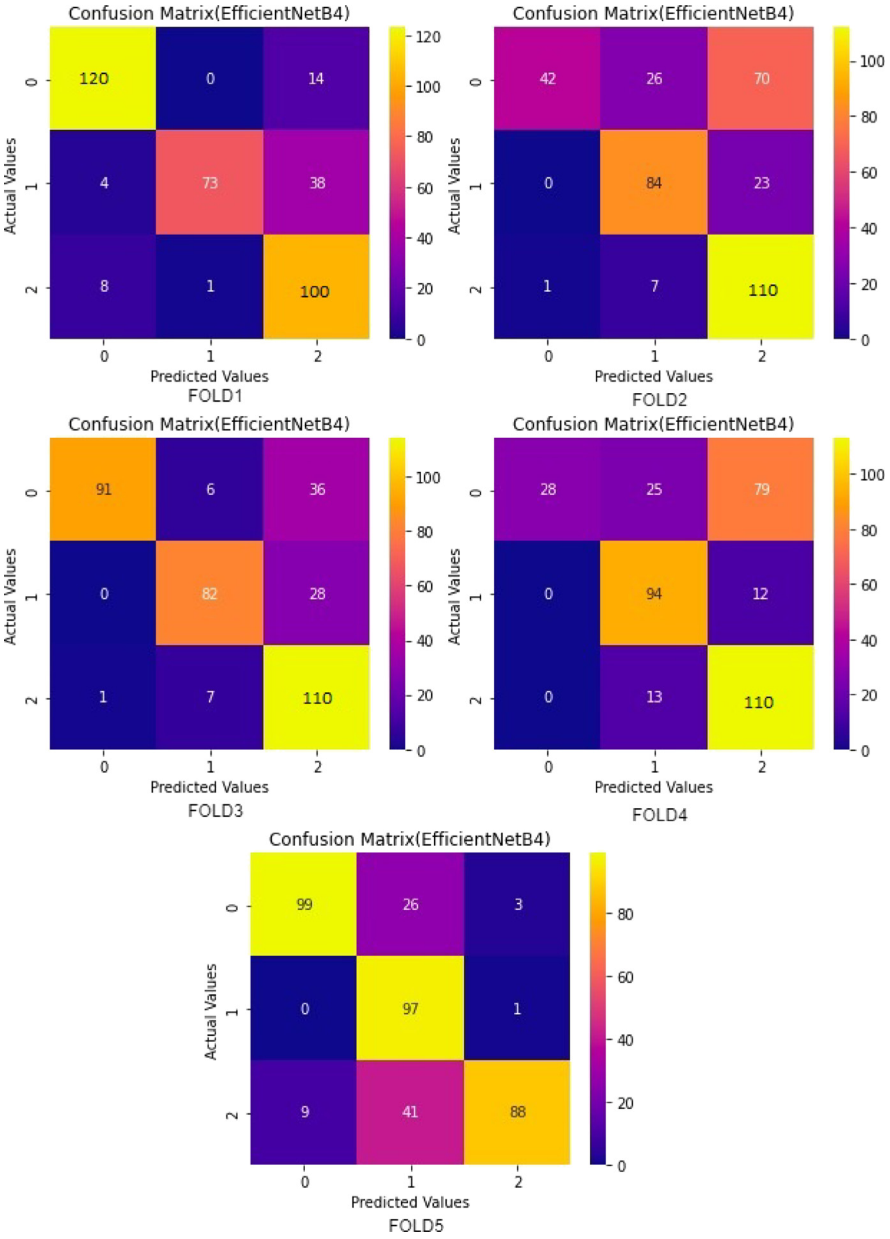


Fig. 6. Confusion Matrix:EfficientNetB4 on 5-Folds(0-Normal, 1-COVID19,2-Viral).

method. It was found that three clusters gave the best results out of all 20. The shuffled dataset was passed through the penultimate or second-last layer of the InceptionV3 functional model, and the extracted results were flattened and appended to a features variable. An instance of K-means clustering with three clusters was declared, and the features variable was fit using the same. The cluster labels were extracted and then compared with the original label data to compute the accuracy. In the semi-supervised method, we fed the training data through the best performing DNN i.e. Inception V3 and for each training dataset, we extracted the features of the penultimate layer. Then, k-means clustering was performed and k-clusters were extracted. The initial accuracy obtained was about 95% with the default hyperparameters when test data was passed through the clustering algorithm. A number of hyperparameters were tuned, like maximum iterations and tolerance, to further tweak the model. The final accuracy achieved for 3 classes of data is 99.69% when the clusters were plugged in with testing data. Fig. 11 shows the result for K-means clustering and the inception of V3-based semi-supervised learning on Dataset 1.

4.4.4. GradCAM

GradCAM is a form of post-hoc attention, meaning it is a method that has been devised for producing heatmaps by applying it to a pre-trained neural network model. The resultant effect is visual explanations from Deep Networks. The CXR image dataset has been used to train several deep learning models, namely VGG16, VGG19, InceptionV3, DenseNet and EfficientNet B4. As a result, all of them produced results with different levels of accuracy and precision. GradCAM has been used for a crystal clear visualisation of the results achieved from various models. GradCAM results have been laid out side by side in a comparative manner in Fig. 15.

Based on the data, we may infer that the suggested InceptionV3 model has superior accuracy and consistency. This was mostly due to the reduction of losses and the improvement in precision. The precision improved because we switched from using the Sequential CNN model to the transfer learning model. The InceptionV3 Transfer Learning Framework. We have also put the model into an Android application, which is not the case for the majority of transfer learning initiatives.

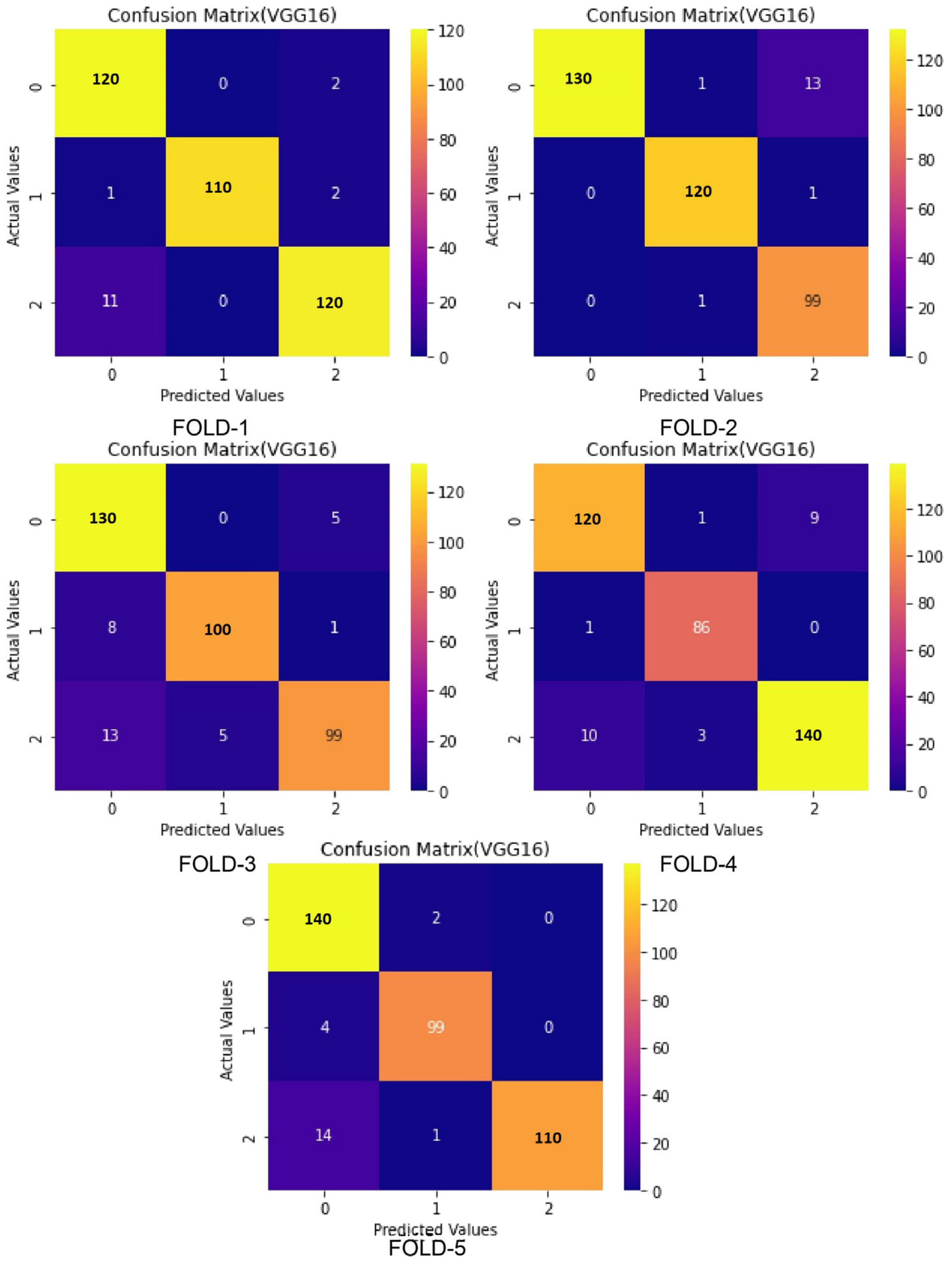


Fig. 7. Confusion Matrix:VGG16 on 5-Folds(0-Normal, 1-COVID19,2-Viral).

While all four transfer learning frameworks performed excellently, when comparing Accuracy and AUC Scores, the InceptionV3 model emerges on top. This is why InceptionV3 was chosen as the foundation for the Android app.

4.5. Cross validation

We used K-fold cross-validation and examined the different aspects of the data to ensure that the CoviDetector model does not overfit or underfit and works effectively. Due to the skewed nature of the data, K-fold (K=5) validation was necessary. At any given time, only one of the five sections of the dataset had been utilised for testing the model, while the others were utilised for training. We have performed the cross-validation on the best-performing model, i.e., InceptionV3. Fig. 16 shows the performance of the InceptionV3 model on various datasets. 5-fold cross-validation is performed for 3 class, 4 class, and 5 class classifications whose confusion matrix is shown in Figs. 12, 13 and 14 respectively.

5. Conclusions and future work

Accurate and timely detection of COVID-19 is necessary in today's world to prevent the further spread of this disease and timely treatment to start. In this study, we describe a method for quickly and easily identify COVID-19 positive patients. DNNs were shown to be effective at separating COVID-19 positive CXR pictures from Normal CXR images. In this paper, four techniques have been adopted, and the best overall has been selected for final classification. With the suggested model, we were able to attain a 99.65% accuracy in our classifications. As an added bonus, a specificity of 1.0 was attained. Pre-trained models can now be easily included in Android apps thanks to technological advancements. Therefore, we turned our focus to COVID-19 detection through Android smartphones. The proposed Android Application has a simple interface to browse through various images and upload one at a time. Once uploaded, the Android Application will be able to classify the image as a COVID-19 or Normal image.

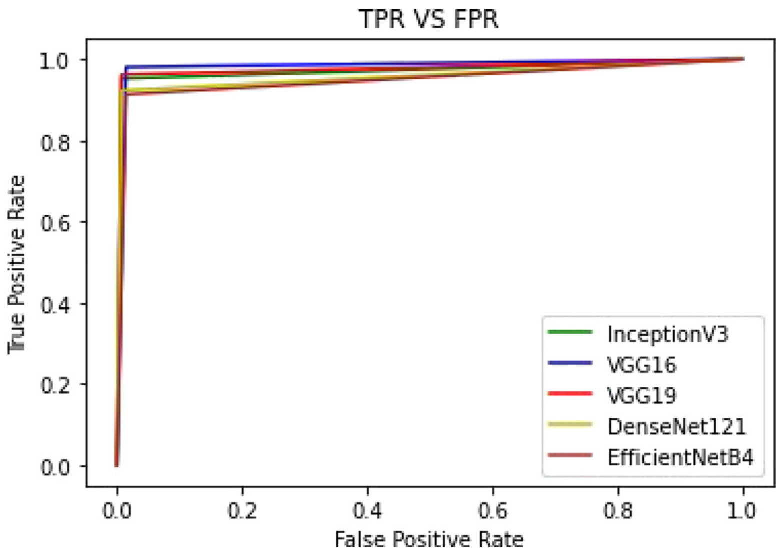


Fig. 8. AUC-ROC curve.

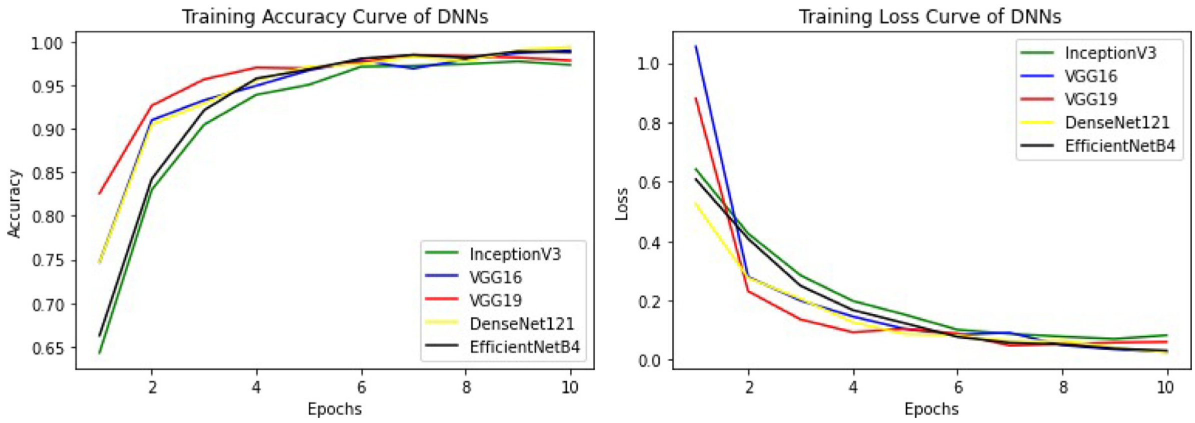


Fig. 9. Accuracy & Loss curves.

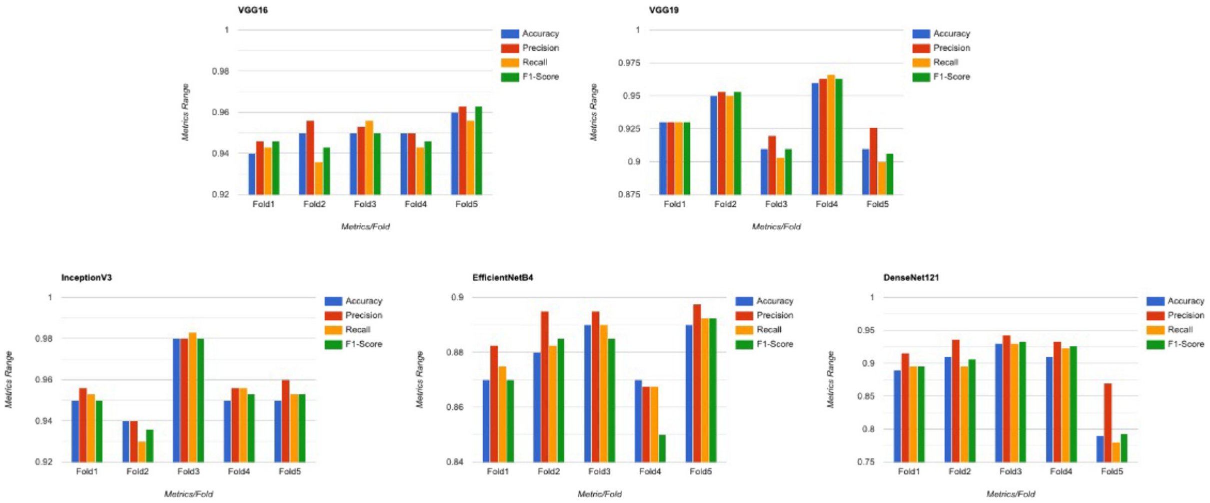


Fig. 10. Metrics chart for all models on Dataset1.

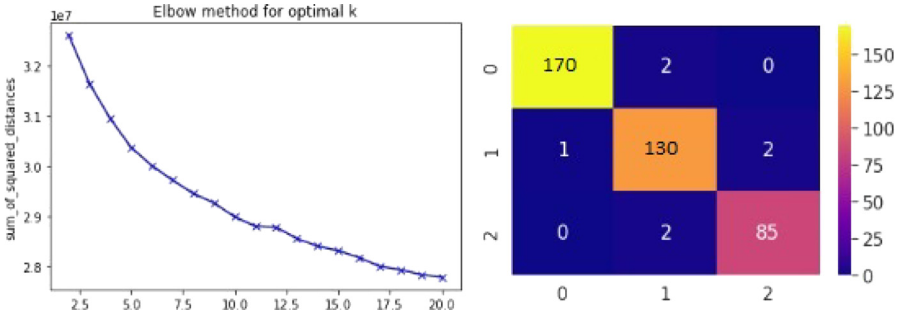


Fig. 11. Result for K-Means Clustering and Inception V3 based semi-supervised learning on Dataset1.

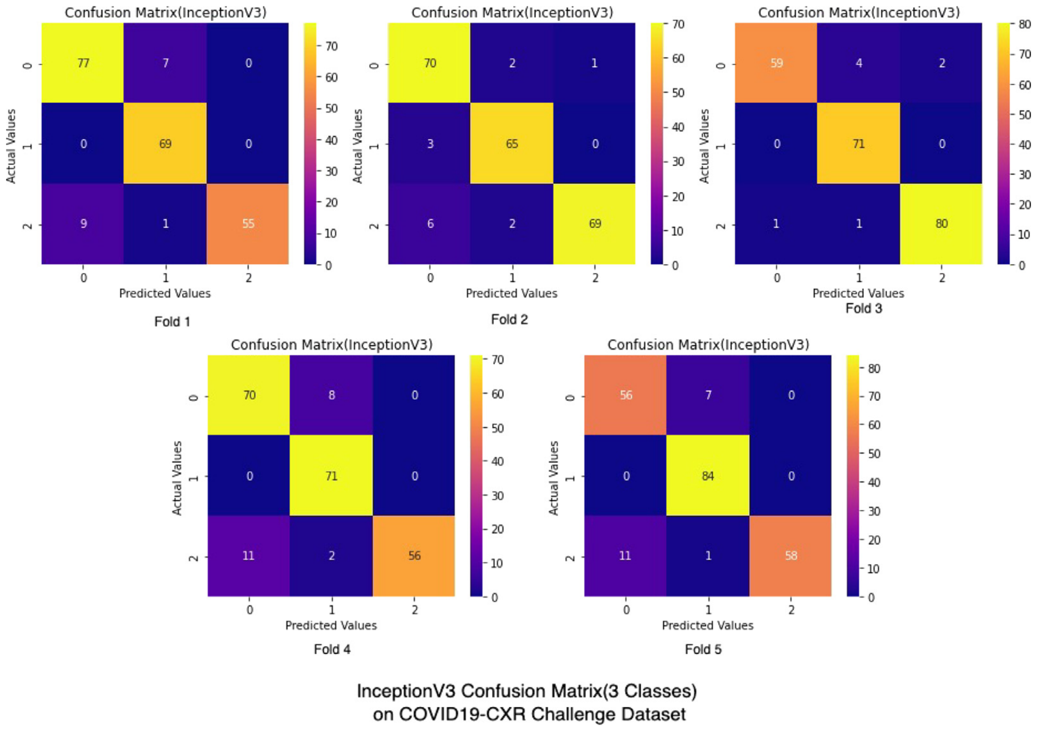


Fig. 12. InceptionV3 results on 3 class Dataset2.

5.1. Possible future directions

In future, CoviDetector can be extended in the following ways:

- **Internet of Things (IoT):** Deep learning algorithms proposed in this paper can be implemented in embedded devices such as the Raspberry Pi or Arduino and can be further used for building the smart X-ray-based COVID-19 detection [52]. CoviDetector also allows for the integration of ChatGPT and IoT, both of which may speed up and enhance patient care. Together, they form a formidable force that is altering the way we engage with technological advances and, perhaps, will make our lives better in generations and decades thereafter [53].
- **Web App:** Web application can be developed which will use this proposed deep learning framework in its backend for predicting the COVID-19 [54].
- **Artificial Intelligence (AI):** Advanced AI methods like quantum machine learning can be used to increase the accuracy rate of detection of COVID-19 [55].
- **Edge AI:** Since latency is a problem in mobile-based applications, we will utilise Edge AI in the future to develop an intelligent framework that will offload the latency-sensitive user requests to the edge node using the latest AI models without any further delay [56].
- **Security:** The CoviDetector itself has no built-in security protections, however, a cryptographic security mechanism might be added in the future to safeguard sensitive information [57].

Software availability

We released CoviDetector available for free as open source. All code, datasets, and result reproducibility scripts are publicly available and can be accessed from GitHub: https://github.com/dasanik2001/CoviDetector

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.

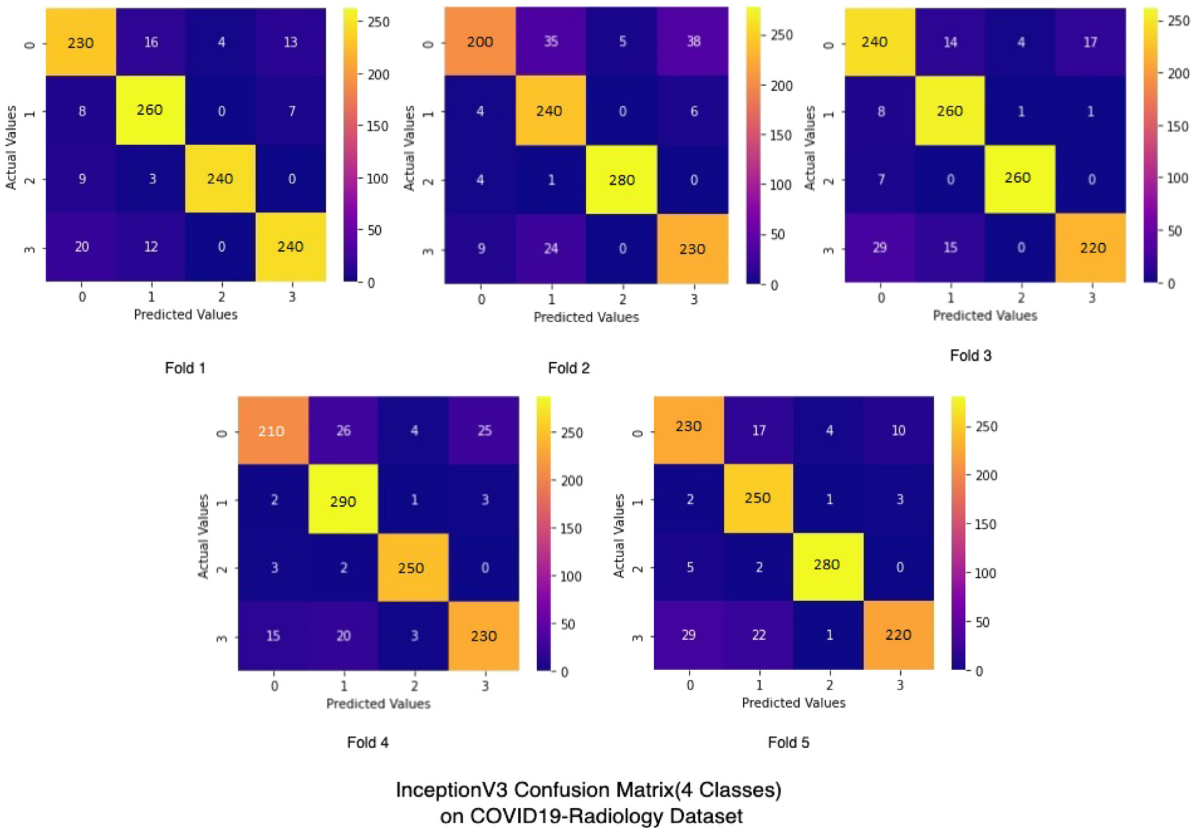


Fig. 13. InceptionV3 results on 4 class Dataset3.

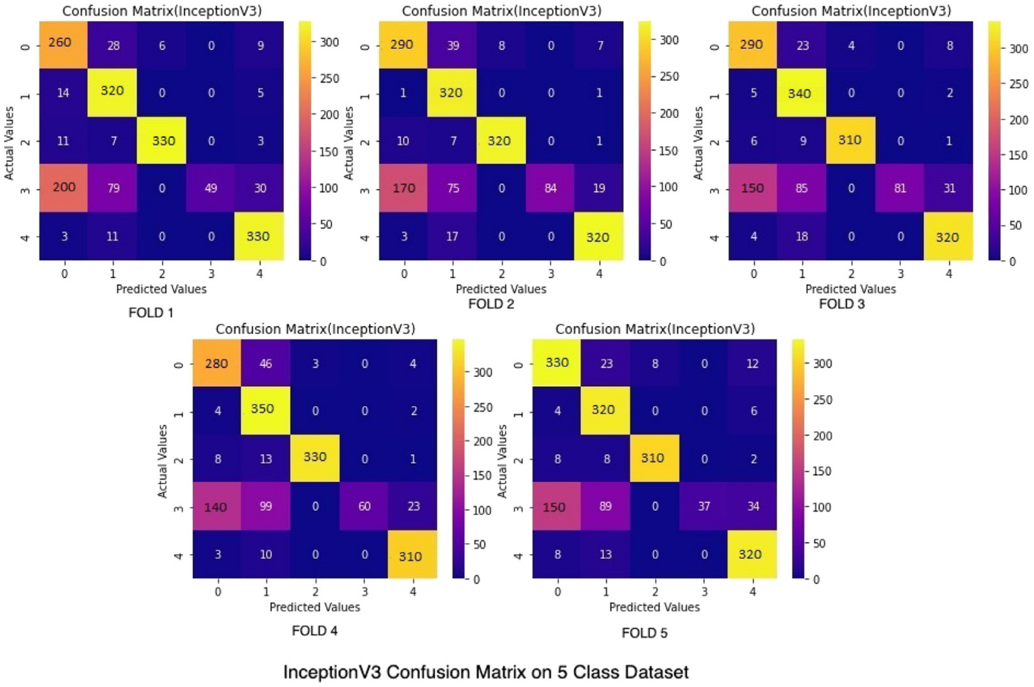


Fig. 14. InceptionV3 results on 5 class Dataset4.

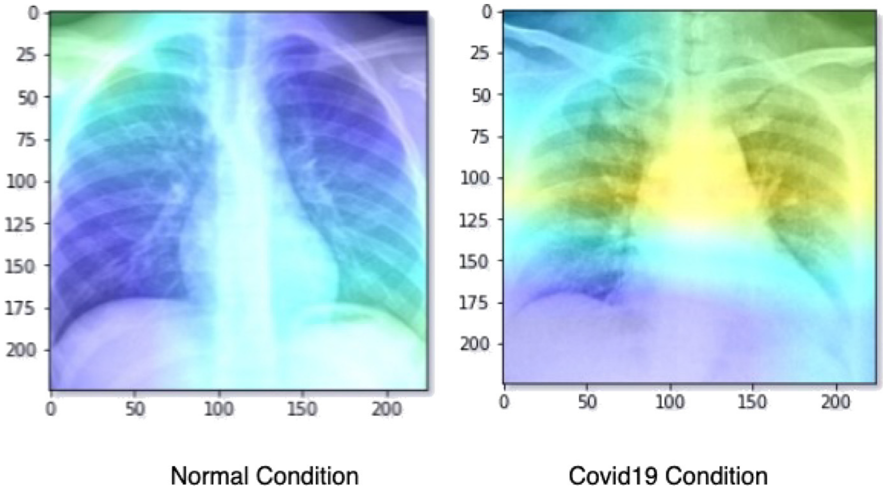


Fig. 15. GradCAM visualisation of InceptionV3-based model on COVID19 and normal condition CXR images.

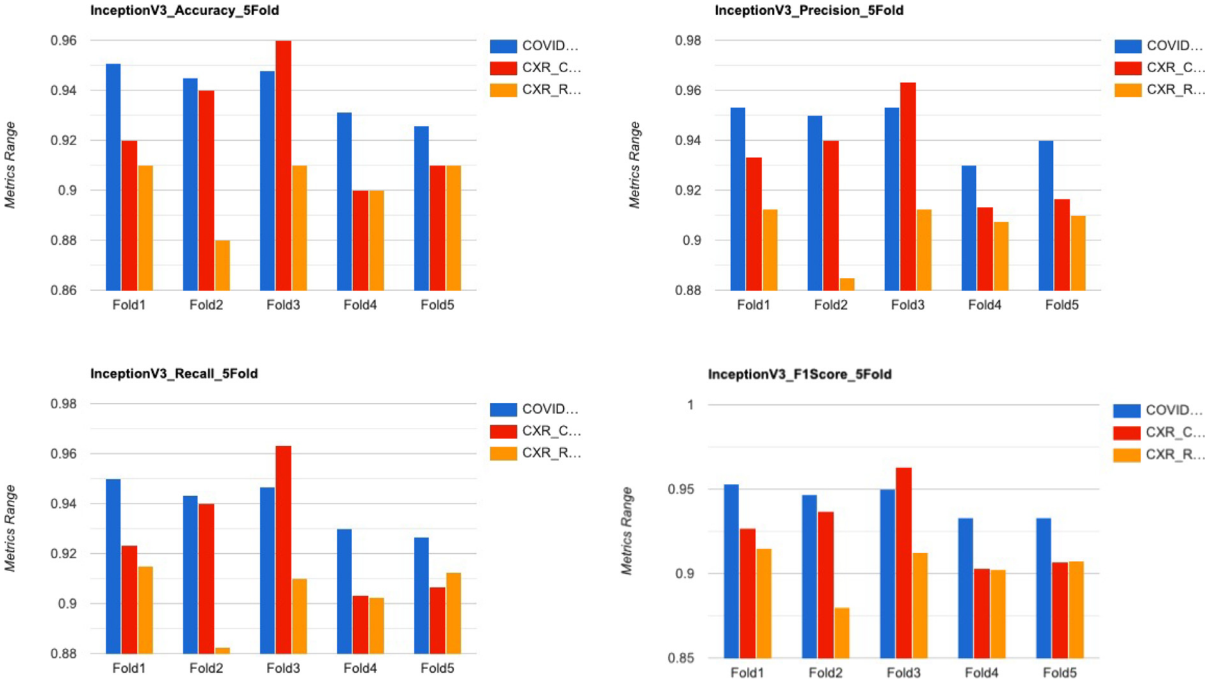


Fig. 16. Comparison of InceptionV3 on various datasets.

References

[1] Who Coronavirus (COVID-19) Dashboard, World Health Organization.

[2] Y. Huang, et al., Training, testing and benchmarking medical AI models using clinical aibench, BenchCouncil Trans. Benchmarks Stand. Eval. 2 (1) (2022) 100037.

[3] Omicron Variant: What You Need to Know, Centers for Disease Control and Prevention.

[4] F. Wu, et al., A new coronavirus associated with human respiratory disease in China, Nature 579 (2020) 1–8.

[5] A. Kumar, K. Sharma, et al., A drone-based networked system and methods for combating coronavirus disease (COVID-19) pandemic, Future Gener. Comput. Syst. 115 (2021) 1–19.

[6] M. Ahsan, et al., COVID-19 detection from chest X-ray images using feature fusion and deep learning, Sensors 21 (2021).

[7] A. Saygılı, A new approach for computer-aided detection of coronavirus (COVID-19) from CT and X-ray images using machine learning methods, Appl. Soft Comput. 105 (2021) 107323.

[8] A. Saygılı, Computer-aided detection of COVID-19 from CT images based on Gaussian mixture model and Kernel support vector machines classifier, Arab. J. Sci. Eng. 47 (2) (2022) 2435–2453.

[9] F. Desai, et al., HealthCloud: A system for monitoring health status of heart patients using machine learning and cloud computing, Internet Things 17 (2022) 100485.

[10] S. Tuli, et al., HealthFog: An ensemble deep learning based smart healthcare system for automatic diagnosis of heart diseases in integrated IoT and fog computing environments, Future Gener. Comput. Syst. 104 (2020) 187–200.

[11] K. Bansal, et al., DeepBus: Machine learning based real time pothole detection system for smart transportation using IoT, Internet Technol. Lett. 3 (3) (2020) e156.

[12] S. Tuli, et al., Predicting the growth and trend of COVID-19 pandemic using machine learning and cloud computing, Internet Things 11 (2020) 100222.

[13] M.F. Aslan, et al., CNN-based transfer learning–BiLSTM network: A novel approach for COVID-19 infection detection, Appl. Soft Comput. 98 (2021) 106912.

[14] D. Kollias, A. Arsenos, S. Kollias, A deep neural architecture for harmonizing 3-D input data analysis and decision making in medical imaging, Neurocomputing 542 (2023) 126244.

[15] A. Arsenos, D. Kollias, S. Kollias, A large imaging database and novel deep neural architecture for COVID-19 diagnosis, in: 2022 IEEE 14th Image, Video, and Multidimensional Signal Processing Workshop, IVMSP, IEEE, 2022, pp. 1–5.

[16] I. Apostolopoulos, M. Tzani, COVID-19: Automatic detection from X-ray images utilizing transfer learning with convolutional neural networks, Australas. Phys.

Eng. Sci. Med. 43 (2020) Supported By the Australasian College of Physical Scientists in Medicine and the Australasian Association of Physical Sciences in Medicine.

[17] P. Verma, et al., FCMCPS-COVID: AI propelled fog–cloud inspired scalable medical cyber-physical system, specific to coronavirus disease, Internet Things 23 (2023) 100828.

[18] M. Golec, et al., HealthFaaS: AI based smart healthcare system for heart patients using serverless computing, IEEE Internet Things J. (2023).

[19] M. Singh, et al., Quantifying COVID-19 enforced global changes in atmospheric pollutants using cloud computing based remote sensing, Remote Sens. Appl. Soc. Environ. 22 (2021) 100489.

[20] M.M. Islam, et al., Diagnosis of COVID-19 from X-rays using combined CNN-RNN architecture with transfer learning, BenchCouncil Trans. Benchmarks Stand. Eval. 2 (4) (2022) 100088.

[21] M. Golec, et al., IFaaSBus: A security-and privacy-based lightweight framework for serverless computing using IoT and machine learning, IEEE Trans. Ind. Inform. 18 (5) (2021) 3522–3529.

[22] D. Mittal, A deep learning approach to detect COVID-19 coronavirus with X-ray images, Biocybern. Biomed. Eng. 40 (2020).

[23] 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) 109761.

[24] K. Ahammed, et al., Early detection of coronavirus cases using chest X-ray images employing machine learning and deep learning approaches, 2020.

[25] M. Azemin, et al., COVID-19 deep learning prediction model using publicly available radiologist-adjudicated chest X-Ray images as training data: Preliminary findings, Int. J. Biomed. Imaging 2020 (2020) 1–7.

[26] T. Ozturk, et al., Automated detection of COVID-19 cases using deep neural networks with X-ray images, Comput. Biol. Med. 121 (2020).

[27] I. Khan, N. Aslam, A deep-learning-based framework for automated diagnosis of COVID-19 using X-ray images, Information 11 (2020) 419.

[28] X. Wang, et al., A weakly-supervised framework for COVID-19 classification and lesion localization from chest CT, IEEE Trans. Med. Imaging PP (2020) 1.

[29] Y. Oh, S. Park, J. Ye, Deep learning COVID-19 features on CXR using limited training data sets, IEEE Trans. Med. Imaging PP (2020) 1.

[30] M. Mohamed, et al., COVID-19 detection from chest X-ray images using artificial-intelligence-based model imported in a mobile application, 2021.

[31] K. Bushra, et al., Automated detection of COVID-19 from X-ray images using CNN and android mobile, Res. Biomed. Eng. 37 (2021).

[32] M. Taresh, et al., Transfer learning to detect COVID-19 automatically from X-ray images using convolutional neural networks, Int. J. Biomed. Imaging 2021 (2021) 1–9.

[33] M. Ahsan, et al., COVID-19 detection from chest X-ray images using feature fusion and deep learning, Sensors 21 (2021).

[34] D.-P. Fan, et al., Inf-Net: Automatic COVID-19 lung infection segmentation from CT scans, 2020.

[35] M. Loey, et al., Within the lack of chest COVID-19 X-ray dataset: A novel detection model based on GAN and deep transfer learning, Symmetry 12 (2020) 651.

[36] N. Wang, et al., Deep learning for the detection of COVID-19 using transfer learning and model integration, 2020, pp. 281–284.

[37] T. Mahmud, et al., 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.

[38] S. Minaee, et al., Deep-COVID: Predicting COVID-19 from chest X-ray images using deep transfer learning, Med. Image Anal. 65 (2020) 101794.

[39] S. Tabik, et al., COVIDGR dataset and COVID-sdnet methodology for predicting COVID-19 based on chest X-ray images, IEEE J. Biomed. Health Inf. 24 (2020) 3595–3605.

[40] J. Xiao, et al., Application of a novel and improved VGG-19 network in the detection of workers wearing masks, J. Phys. Conf. Ser. 1518 (2020) 012041.

[41] S. Tammina, Transfer learning using VGG-16 with deep convolutional neural network for classifying images, Int. J. Sci. Res. Publ. (IJSRP) 9 (2019) p9420.

[42] S. Wang, Y.-D. Zhang, DenseNet-201-based deep neural network with composite learning factor and precomputation for multiple sclerosis classification, ACM Trans. Multimed. Comput. Commun. Appl. 16 (2020) 1–19.

[43] K. Boonyuen, et al., Convolutional neural network inception-v3: A machine learning approach for leveling short-range rainfall forecast model from satellite image, 2019, pp. 105–115.

[44] M. Tan, Q. Le, Efficientnet: Rethinking model scaling for convolutional neural networks, in: International Conference on Machine Learning, PMLR, 2019, pp. 6105–6114.

[45] M. Siddhartha, A. Santra, COVIDLite: A depth-wise separable deep neural network with white balance and CLAHE for detection of COVID-19, 2020.

[46] D. Kermany, et al., Identifying medical diagnoses and treatable diseases by image-based deep learning, Cell 172 (2018) 1122–1131.e9.

[47] A. Tahir, et al., COVID-19 infection localization and severity grading from chest X-ray images, 2021.

[48] M. Chowdhury, et al., Can AI help in screening Viral and COVID-19 pneumonia? IEEE Access 8 (2020) 132665–132676.

[49] D. Kollias, A. Tagaris, A. Stafylopatis, S. Kollias, G. Tagaris, Deep neural architectures for prediction in healthcare, Complex Intell. Syst. 4 (2018) 119–131.

[50] N. Bouas, Y. Vlaxos, V. Brillakis, M. Seferis, S. Kollias, Deep transparent prediction through latent representation analysis, 2020, arXiv preprint arXiv:2009.07044.

[51] Y. Vlaxos, M. Seferis, S. Kollias, Transparent adaptation in deep medical image diagnosis, in: Trustworthy AI-Integrating Learning, Optimization and Reasoning: First International Workshop, TAILOR 2020, Virtual Event, September 4–5, 2020, Revised Selected Papers 1, Springer, 2021, pp. 251–267.

[52] T. Shao, et al., IoT-Pi: A machine learning-based lightweight framework for cost-effective distributed computing using IoT, Internet Technol. Lett. (2022) e355.

[53] S.S. Gill, R. Kaur, ChatGPT: Vision and challenges, Internet Things Cyber-Phys. Syst. 3 (2023) 262–271.

[54] D. Chowdhury, et al., Covacdiser: A machine learning-based web application to recommend the prioritization of COVID-19 vaccination, in: Intelligence Enabled Research, Springer, 2022, pp. 105–117.

[55] S.S. Gill, et al., AI for next generation computing: Emerging trends and future directions, Internet Things (2022) 100514.

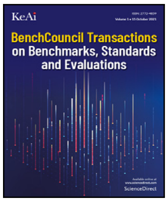
[56] R. Singh, et al., Edge AI: a survey, Internet Things Cyber-Phys. Syst. 3 (2023).

[57] Y. Zhou, et al., An efficient encrypted deduplication scheme with security-enhanced proof of ownership in edge computing, BenchCouncil Trans. Benchmarks Stand. Eval. 2 (2) (2022) 100062.

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/



Full length article

DPUBench: An application-driven scalable benchmark suite for comprehensive DPU evaluation

Zheng Wang [a,b], Chenxi Wang [a,b,*], Lei Wang [a,b]

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

ARTICLE INFO

Keywords:
DPU
Evaluation
Benchmark
Application-driven

ABSTRACT

With the development of data centers, network bandwidth has rapidly increased, reaching hundreds of Gbps. However, the network I/O processing performance of CPU improvement has not kept pace with this growth in recent years, which leads to the CPU being increasingly burdened by network applications in data centers. To address this issue, Data Processing Unit (DPU) has emerged as a hardware accelerator designed to offload network applications from the CPU. As a new hardware device, the DPU architecture design is still in the exploration stage. Previous DPU benchmarks are not neutral and comprehensive, making them unsuitable as general benchmarks. To showcase the advantages of their specific architectural features, DPU vendors tend to provide some particular architecture-dependent evaluation programs. Moreover, they fail to provide comprehensive coverage and cannot adequately represent the full range of network applications. To address this gap, we propose an **application-driven** scalable benchmark suite called **DPUBench**. DPUBench classifies DPU applications into three typical scenarios — network, storage, and security, and includes a scalable benchmark framework that contains essential Operator Set in these scenarios and End-to-end Evaluation Programs in real data center scenarios. DPUBench can easily incorporate new operators and end-to-end evaluation programs as DPU evolves. We present the results of evaluating the NVIDIA BlueField-2 using DPUBench and provide optimization recommendations. DPUBench are publicly available from https://www.benchcouncil.org/DPUBench.

1. Introduction

In the past decade, the growth rate of CPU performance has been relatively slow due to the physical limitations it faces [2]. As the size of transistor circuits approaches the scale of atoms, increasing challenges caused by physical limitations, such as leakage, have led to the failure of Dennard Scaling Law [3]. In contrast, many emerging computing fields, such as artificial intelligence (AI), big data, and the Internet of Things, are thriving as computing resources reach a threshold scale. The demand for computing resources in these fields is rapidly growing, resulting in CPU becoming increasingly incapable of meeting it in data centers. As a result, deploying specialized chips, such as GPU, TPU [4], and DPU, in data centers has become a new trend for both academia and industry.

DPU is a hardware accelerator designed to offload network applications from CPU in data centers. With the increase in network bandwidth from 10 Gbps to 25 Gbps, 40 Gbps, 100 Gbps, 200 Gbps, and even 400 Gbps, CPU has become increasingly burdened by network applications, and its computing resources are heavily consumed before processing computing applications. To ensure that CPU's computing resources are focused on CPU-bound applications, DPU has emerged.

DPU typically consists of multiple hardware accelerators for network applications, a multi-core CPU for scheduling and programming, and high-bandwidth network IO interfaces [5]. As an emerging hardware accelerator, the DPU architecture has not yet been standardized and is decided by DPU manufacturers. Typical DPU architectures include those that can fully offload infrastructural network applications in data centers, such as NVIDIA Bluefield [6]; those that offload specific network application scenarios in data centers, such as YUSUR KPU [7–10]; and programmable architectures developed based on FPGA, such as Intel Mount Evans [11].

Benchmarking is a widely-used research method in computer science for evaluating the performance of systems. Benchmarking evaluations can provide insights into the actual performance of the evaluated object and can guide future co-design and optimization of software and hardware. With the development of DPU and data centers, a DPU benchmark suite is necessary. However, to the best of our knowledge, there is currently no benchmark suite available for comprehensive

* Corresponding author.
E-mail addresses: wz917942636@gmail.com (Z. Wang), wangchenxi21s@ict.ac.cn (C. Wang), wanglei_2011@ict.ac.cn (L. Wang).

https://doi.org/10.1016/j.tbench.2023.100120
Received 2 June 2023; Received in revised form 4 July 2023; Accepted 5 July 2023
Available online 7 July 2023

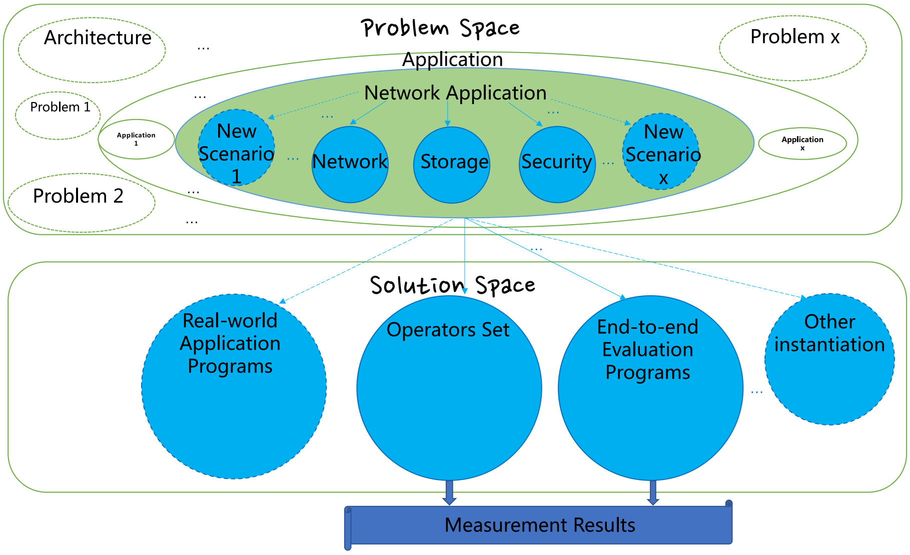


Fig. 1. The overview of DPUBench. Inspired by Zhan's [1] benchmarking methodology, DPUBench comprises problem definition, instantiation, and measurement. The solid lines in the figure indicate the current implementations of DPUBench, while the dashed lines indicate future implementations that can be added or other benchmark implementations. The elliptical box represents the problem definition, the thin arrow, and circle denote the specific instantiation, and the thick arrow at the bottom represents the measurement results output of DPUBench.

DPU evaluation. Existing DPU evaluation programs are either provided by DPU manufacturers [7–10,12,13], which are based on their specific DPU architecture design, or they are selected and rewritten from some open source benchmark programs based on the architecture characteristics of the evaluated DPU in previous research [14–19]. These previous DPU evaluation programs are architecture-dependent, meaning that they are designed based on a specific architecture and not suitable to evaluate DPU with different architectures. To perform a comprehensive DPU evaluation, the architecture-dependent DPU benchmark programs are not feasible at this stage because the DPU architecture has not yet been standardized and is undergoing rapid evolution. Even evolving DPU architectures from the same manufacturer may have significant differences.

Zhan [1] proposed a benchmarking methodology from the problem definition, instantiation, and measurement, making benchmark design and research more standardized and theoretical. We utilized Zhan's methodology to develop our benchmark suite, DPUBench and adopted an application-driven approach at the problem definition stage. For problem instantiation, we selected network, storage, and security as the typical DPU application scenarios. At the solution instantiation stage, we constructed operators in these scenarios to evaluate early DPU designs and developed end-to-end evaluation programs to obtain results in a real data center environment. DPUBench is an application-driven scalable benchmark framework that can easily incorporate new operators and end-to-end evaluation programs as DPU evolves. As an application-driven benchmark, DPUBench can add new DPU application scenarios and corresponding operators and end-to-end evaluation programs, regardless of any changes to the DPU architecture. A DPUBench overview is presented in Fig. 1.

Our contributions are as follows.

(1) We present DPUBench, an application-driven scalable benchmark suite for comprehensive DPU evaluation. DPUBench is scalable and standardized, which can accommodate new operators and end-to-end evaluation programs as DPU architecture evolves, making it a comprehensive and fair benchmark suite for DPU evaluation.

(2) We select network, storage, and security as typical DPU application scenarios and extract 16 representative operators from real-world applications. Our experiments demonstrate that these operators have good representativeness, diversity, and coverage, making them suitable for low-cost evaluation of early-stage DPU designs.

(3) We develop two end-to-end DPU workloads for typical DPU applications and measure their throughput, packet loss ratio, Server CPU utilization ratio and latency in a real data center machine. Our experiments demonstrate the effectiveness of end-to-end evaluation programs in assessing the performance of DPUs in real-world network applications.

(4) We evaluate NVIDIA BlueField-2 [6] using DPUBench and provide optimization recommendations. Our experiments reveal that NVIDIA BlueField-2 can efficiently offload network applications from the CPU, particularly network storage protocol and DPI applications. In the end-to-end evaluation, we demonstrate that NVIDIA BlueField-2 can effectively reduce the server CPU utilization ratio in network applications and allocate more CPU computing resources for computing applications.

The rest of this paper is structured as follows: Section 2 provides the background and motivation. Section 3 introduces the methodology of DPUBench. Section 4 presents the Operator Set in DPUBench and the corresponding experimental results. Section 5 discusses the End-to-end Evaluation Programs in DPUBench along with their respective experiment results. Section 6 concludes with a discussion of related work, while Section 7 outlines the conclusions and plans for further work.

2. Background and motivation

In this section, we will first introduce the background of DPUBench, including existing DPU benchmarks, DPU evaluation programs, and DPU characterization studies. Next, we will have a brief introduction to the NVIDIA BlueField-2 DPU. Based on the above discussion, we will provide the motivation for DPUBench.

2.1. Background of DPUBench

DPU is a new hardware accelerator in data centers designed to offload network applications from the CPU. However, due to the lack of standardized DPU architectures, DPU evaluation is typically conducted by DPU manufacturers who provide evaluation programs that are

Table 1
The overview of representative existing DPU evaluation studies.

Benchmark/Programs	Provider	Workload	Metric	Evaluation object
RXPBench [12]	NVIDIA	Regular Expression Matching	Time	NVIDIA BlueField DPU
Evaluation Programs [13]	Liguori	Hypervisor	Performance Metric	Amazon Nitro DPU
Evaluation Programs [7–10]	YUSUR	SQL Program	Latency	YUSUR DPU
Evaluation Programs [14]	Wei	RDMA Read/Write & Send/Recv Request	End-to-end Latency Throughput Bottleneck	NVIDIA BlueField-2
Evaluation Programs [15]	Ibanez	RPC Program	Wire-to-wire Latency Throughput	RPC SmartNIC
Evaluation Programs [16]	Ma	Matrix Multiplication	Time Training Time	AI SmartNIC
Evaluation Programs [17]	Mandal	RDMA Read/Write	Throughput	Storage SmartNIC
Evaluation Programs [18]	Sabin	RDMA Read/Write	Throughput	Security SmartNIC
Evaluation Programs [19]	Bosshart	Network Transport Protocol	Latency	SDN SmartNIC

tailored to their own products or by researchers who select and rewrite existing benchmark programs based on the architectural characteristics of a specific DPU product. Currently, there is no benchmark suite available for comprehensive DPU evaluation. Table 1 has summarized some representative DPU evaluation studies from previous work.

From Table 1, we observe that all of the DPU evaluation programs [7–10,12–19] listed are designed for DPUs with similar architecture or for SmartNICs with specific acceleration units. Moreover, eight out of nine of these programs are [7–10,12,13,15–19] designed for one single specific application scenario, which results in inadequate coverage for comprehensive DPU evaluation. The lack of evaluation programs in a real network environment also limits the efficacy and reliability of the results. Three out of nine of these programs [12,13,16] are not conducted in a real network environment in data centers, further limiting their relevance to real-world network applications evaluation. Additionally, three out of nine of these programs [12,13,16] only measure performance metrics commonly used for CPU evaluation, which are insufficient for DPU evaluation as they do not take into account network-related metrics such as network throughput and latency.

2.2. NVIDIA BlueField-2 DPU

NVIDIA BlueField-2 is a typical DPU that is designed to fully offload infrastructural network applications in data centers. Its architecture, as shown in Fig. 2, integrates a variety of hardware acceleration units for network applications, high bandwidth network IO interfaces, a multi-core ARM AArch64 processor, and optional on-board DRAM of either 16 GB or 32 GB [6]. The hardware acceleration units of NVIDIA BlueField-2 can help offload infrastructural network application operators, such as the regular expression matching acceleration unit (Reg-Ex) for regular expression matching operator and the public key encryption and decryption acceleration unit (Public-Key Crypto) for public key encryption and decryption operator. The high bandwidth network IO interfaces include the ConnectX interface with two 100 Gbps Remote Direct Memory Access (RDMA) [20] ports or one single 200 Gbps Ethernet/InfiniBand [21] port, which are used in production environments. The ARM AArch64 processor contains 8 Cortex-A72 cores with a 2.75 GHz frequency, sharing a 4 MB L2 Cache between cores and an 8MB L3 Cache among the units of NVIDIA BlueField-2. As for memory units, NVIDIA BluField2 equips with DDR4-1600 DRAM and eMMC flash memory, which are used for storage that will not be lost after a power failure.

2.3. Motivation of DPUBench

Section 2.1 has provided a brief introduction of representative existing DPU benchmarks or evaluation programs, all of which are used for one specific DPU or DPU with one specific architecture. However, there is currently no DPU benchmark that can effectively evaluate DPUs with different architectures, which is a significant gap in the field. Furthermore, DPU architecture is rapidly evolving due to the increasing CPU computing resources that need to be offloaded in data centers, resulting in significant differences in DPU architectures between different DPU manufacturers or even the adjacent generations of the same manufacturer. Therefore, a scalable DPU benchmark that can evaluate DPUs of different architectures is needed, and it should be able to support the addition of new evaluation programs and metrics to accommodate the rapid development of DPUs.

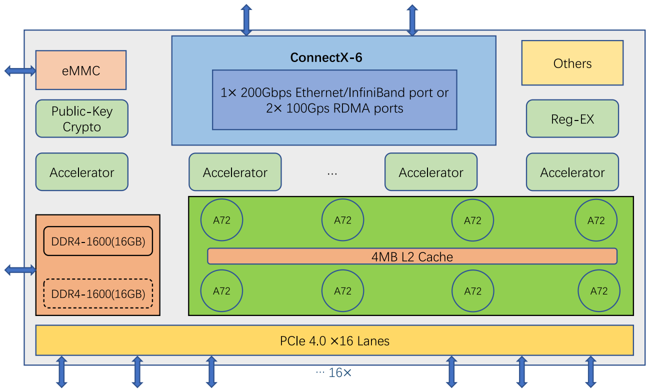


Fig. 2. The architecture of NVIDIA BlueField-2.

Another motivation behind DPUBench is to ensure the representativeness and coverage of the benchmark suite, as well as the effectiveness and reliability of the evaluation results. In terms of coverage, the benchmark programs should not only be at a certain scale to handle basic network application scenarios but also not impose excessive evaluation costs in terms of time and resource utilization. Additionally, to ensure the reliability of the evaluation results, network-related metrics should be carefully selected, and DPUs should be evaluated in a real network environment within data centers.

3. Methodology

A study is typically aimed at solving a specific problem or class of problems with corresponding solutions. To construct DPUBench, we divide the process into problem space and solution space and implement it step by step in these two spaces. Our methodology for DPUBench is illustrated in Fig. 1, which includes problem definition, problem instantiation, solution instantiation, and measurement results. This methodology is inspired by Zhan's benchmark science methodology [1] of problem definition, instantiation, and measurement. By providing a clear and detailed description of each step in the construction of DPUBench, our methodology makes it easy to develop, maintain, and update. In this section, we will provide a detailed introduction to the various steps involved in constructing DPUBench.

3.1. Problem definition

The first step in constructing DPUBench is problem definition, which involves clarifying the research object and establishing a clear research direction. The problem definition is critical in the problem space as there are numerous problems, and failure to define the problem may lead to deviations or even irrelevance in the final research results. For example, DPU is used to offload network applications from CPU in data centers, so the evaluation of DPU should focus on network application issues. Without proper problem definition, the evaluation metrics may be directly determined as performance metrics, such as time, resulting in evaluation results that do not accurately reflect the DPU's capabilities.

The problem definition of DPUBench aims to determine the construction of the benchmark suite from an application perspective, specifically for network applications. In this regard, network-related metrics such as latency and throughput are selected as the evaluation metrics of DPUBench, and the throughput acceleration ratio is chosen as the performance metric, along with the CPU utilization ratio. Other approaches for problem definition in the problem space for conducting a DPU benchmark include determining the construction of the benchmark suite from an architecture perspective, a simulation perspective, and a real object perspective, among others.

However, due to the rapidly evolving nature of DPU architecture and products, adopting an application-driven benchmark construction as the problem definition of DPUBench methodology is more comprehensive, clear, and easy to expand and update while maintaining the authenticity and effectiveness of DPU evaluation. Since DPU is used to offload network applications from the CPU in data centers, developing a benchmark suite with a focus on network applications provides a stable platform for DPU evaluation, as network application development is in a relatively stable stage compared to the rapidly iterating DPU architecture.

3.2. Problem instantiation

After the problem definition, the next step in constructing DPUBench is problem instantiation. This involves concretizing the defined problem within a certain scope, thereby transforming research from abstract theory into concrete practice. Different researchers may approach the same defined problem from different perspectives and research different aspects of it. Even the same research team may have different understandings of the problem at different stages of research, resulting in differences in the research focus. Problem instantiation serves to unify the specific boundaries of the research problem after the problem definition and before the solution instantiation. This makes subsequent research solutions more standardized and unified, with a clear methodology roadmap.

The problem instantiation of DPUBench involves selecting network, storage, and security as typical network application scenarios and implementing DPUBench based on these three scenarios. These scenarios are chosen based on previous work [7–10,12–15,17–19], which identifies them as common representative scenarios for DPU at the current stage of offloading network applications from CPU in data centers.

By selecting these three scenarios, DPUBench covers different aspects of network applications. The network scenario covers various network transmission protocols, the storage scenario includes compression and decompression algorithms as well as storage protocols, and the security scenario encompasses various encryption and decryption algorithms. As a result, DPU evaluation with DPUBench is more comprehensive.

To maintain focus on the network applications, we do not select AI or other computation-intensive scenarios as representative scenarios, as only a few DPUs [16,22] can assist with those scenarios at the current stage. However, as DPUs and data centers continue to develop, these scenarios may become representative scenarios for network applications in data centers, and we will add them in future versions of DPUBench.

3.3. Solution instantiation

We then do the solution instantiation and implement DPUBench. Solution instantiation is to solve the instantiated problems and provide the research outcomes. It is a critical step in scientific research as it enables the provision of tangible research outcomes, such as tools, products, and research papers. And in DPUBench, solution instantiation is one step of the methodology.

The solution instantiation of DPUBench involves the extraction and implementation of basic operators from network, storage, and security scenarios, which compose the DPUBench's Operator Set for DPU evaluation. We also implement end-to-end evaluation programs to conduct DPU evaluation in a real network environment. Operators represent the most common algorithms in these three scenarios, and their combination can construct typical programs in each scenario. The end-to-end evaluation programs simulate the business of a real data center machine and evaluate the performance of DPU in a real network environment through communication between the Client and Server. We do not include a separate application set in DPUBench because the main execution part of application programs can be implemented with operators combination, and their evaluation cost is higher compared to operators, as well as their evaluation results are less reliable and effective compared to end-to-end evaluation programs.

Table 2 provides a brief summary of the methodology used in DPUBench. And the overview of DPUBench's methodology is shown in Fig. 1, which consists of problem definition, problem instantiation, and solution instantiation.

4. Operator set of DPUBench

Operator Set is a component of the solution instantiation in DPUBench, as mentioned in Section 3. In this section, we will outline the process of extracting the fundamental operators from network, storage, and security scenarios for DPUBench. We will then present the experimental results of the Operator Set, which include validating its representativeness and coverage, as well as evaluating the NVIDIA BlueField-2 using the micro-benchmarks of Operator Set. Based on these evaluation results, we will provide optimization recommendations for utilizing DPU effectively.

4.1. The extraction of operator set of DPUBench

We initially establish two rules for extracting the operators in DPUBench, and then conclude the typical programs and protocols in network, storage, and security scenarios based on previous work [7–10, 12–19] in Table 3 to comply with **Rule1**. Additionally, upon breaking down these programs, we observe that certain processes, such as the three-way handshake in TCP/IP protocol [23] and the establishment of a secure initial key in an IPSec session [24], are executed only once during program initialization and have a relatively small proportion of execution time. Therefore, we do not extract operators from these processes to ensure representativeness. Instead, we decompose the most time-consuming and frequently executed processes within these typical programs to derive the operators for DPUBench based on **Rule2**. The two rules for extracting DPUBench operators are defined as follows.

Rule1. Operators should be integral components of typical programs on the network applications DPU has offloaded.

Rule2. The combinations of operators should constitute the primary execution portion of the typical programs on the network applications DPU has offloaded.

Table 2
The brief summary of DPUBench's methodology. We construct DPUBench from the perspectives of the problem definition, the problem instantiation, the solution instantiation.

Benchmark suite	Problem definition	Problem instantiation	Solution instantiation	
DPUBench	Network Applications	Network Scenario Storage Scenario Security Scenario	Operator Set	End-to-end Evaluation Programs

Table 3
The typical programs and protocols in network, storage, and security scenarios.

Network	TCP/IP [23], RDMA [25], OVS [26]
Storage	VirtIO-Blk [27], NVMe-Of [28]
Security	OpenSSL [29], IPSec [24], IDS

Table 4
The Operator Set of DPUBench.

Network	LPM, TCPSeg, IPSeg, CheckSum, CRC, Toeplitz
Storage	LZ77, Huffman, Snappy, CheckSum
Security	RSA, AES, DSA, ECDSA, MD5, SHA256, LPM, RXPMatch

4.1.1. Operators extraction in network scenario

From Table 3, we start with decomposing the data packet processing of the TCP/IP protocol [23] in network scenario, as it serves as the fundamental protocol used in networking. As illustrated in Fig. 3, the data packet processing of the TCP protocol is decomposed into three parts: TCP fragmentation, TCP checksum, and data copying. Similarly, the data packet processing of the IP protocol is decomposed into four parts: IP fragmentation, IP route lookup, IP checksum, and data copying.

TCP/IP protocol defines a maximum length for data packets to ensure efficient transmission in a network [23]. Therefore, the first step in transmitting a data packet is to perform packet fragmentation, which divides the packet into smaller fragments that do not exceed the maximum length specified in the protocol. From this process, we extract the TCPSeg and IPSeg operators. And in the IP protocol, when processing a data packet, it needs to perform a route lookup in the route table to determine the destination IP address and update the route table accordingly. For this operation, we extract the Longest Prefix Match (LPM) operator. To ensure the integrity of transmitted data packets, TCP/IP protocol uses the Checksum algorithm for verification when the receiving node in the data center receives the data packet. From this process, we extract the CheckSum operator. Since TCP/IP protocol programs run in the kernel space, data packets that need to be transferred typically undergo at least one data copying process. From this operation, we extract the MemCpy operator. However, please note that the MemCpy operator is currently under development and some bugs still need to be fixed.

In the Ethernet protocol, we focus on the data packet reception processing and decompose it into several key operations. As shown in Fig. 4, we extract LPM, CRC and Toeplitz operators from those operations. The Longest Prefix Match (LPM) operator is used for ARP MAC address resolution, which involves looking up the MAC address in the ARP table based on the destination IP address. The Cyclic Redundancy Check(CRC) operator is used for error detection and verification of the received data packet, as well as the Toeplitz operator used for performing a hash map for Receive Side Scaling (RSS) core selection in a multi-core processor.

The operators in the RDMA protocol [25] and Open vSwitch (OVS) protocol [26] are encompassed by the extracted operators in the previous network scenario. All the extracted operators in the network scenario of DPUBench are summarized in Table 4.

4.1.2. Operators extraction in storage scenario

In storage scenario, we focus on the data packet processing in storage protocols such as VirtIO-Blk [27] and NVMe-OF [28]). As shown in Fig. 5, we extract LZ77, Huffman, and Snappy operators. The Lempel–Ziv-77(LZ77) operator is based on a lossless data compression algorithm that achieves compression by replacing repeated occurrences of data with references to a dictionary [30]. The Huffman operator is based on the Huffman coding algorithm [31], which is a variable-length prefix coding technique used for lossless data compression. And the

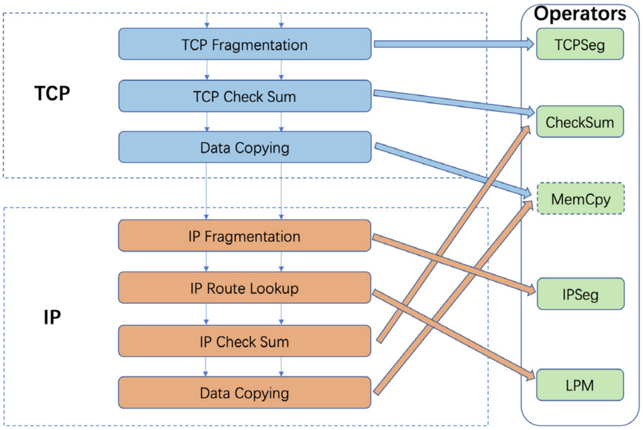


Fig. 3. The operators extracted in TCP/IP protocol.

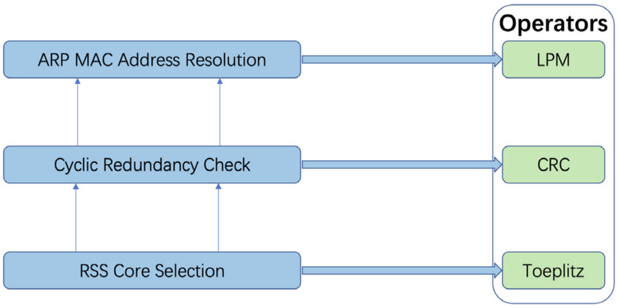


Fig. 4. The operators extracted in Ethernet protocol.

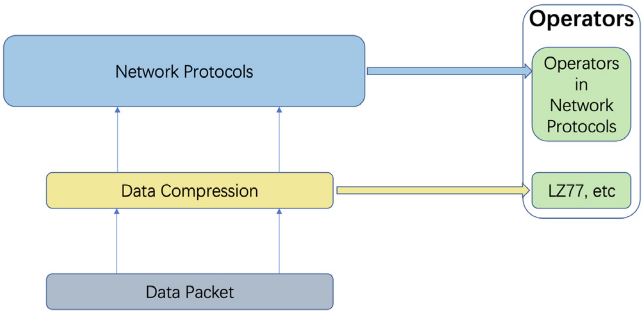


Fig. 5. The operators extracted in Storage protocols.

Table 5
The validation of Rule1 and Rule2 for operator set in DPUBench.

Programs	Scenarios	Operators
L3fwd	Network	LPM, CheckSum, CRC, IPSeg, Toeplitz
IPSec	Network & Security	LPM, RSA, CheckSum, CRC
File-Compress	Storage	Compress
File-Integrity	Network & Security & Storage	SHA256, SHA1, MD5
IPS	Security	RXPMatch, TCPSeg
Url-Filter	Security	RXPMatch

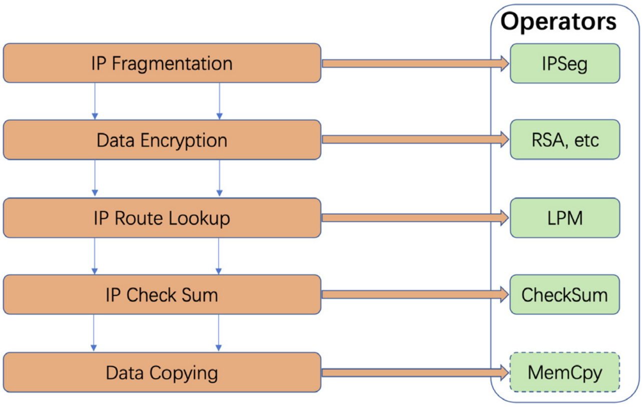


Fig. 6. The operators extracted in IPSec protocol.

Snappy operator is based on a fast, block-based compression algorithm that aims to provide high compression and decompression speeds with reasonable compression ratios.

These compression operators are commonly used in storage scenarios to compress data packets before transmission to effectively utilize network bandwidth. The detailed network protocols and their operators are discussed in Section 4.1.1, while in the storage scenario, we primarily focus on extracting compression operators. All the operators of DPUBench in the storage scenario are summarized in Table 4.

4.1.3. Operators extraction in security scenario

In a security scenario, programs can be primarily classified into network protocols (such as OpenSSL [29] and IPSec [24]) for ensuring data security during transmission, as well as application programs (e.g., Firewall) based on Deep Packet Inspection (DPI). We decompose the IPSec protocol [24] in Fig. 6 and extract four additional encryption operators, in addition to the network operators extracted in Section 4.1.1. We implement RAS [32], AES [33], DSA, and ECDSA [34] operators for data compression. The operators in OpenSSL [29] are already included in the operators extracted from IPSec [24].

The decomposition of DPI is illustrated in Fig. 7. In the network context, data packets undergo regular expression matching before transmission. The outcome of the regular expression matching determines whether the data packet is transmitted over the network. In addition to the operators extracted in the network scenario, we include the RXPMatch operator for performing regular expression matching. Table 4 presents all the operators implemented in DPUBench.

4.2. The experiments of operator set of DPUBench

4.2.1. Experimental configurations

The experiments are conducted on two platforms: the Intel Xeon E5-2620 v3 CPU (with 2 processors), which has 10 GB of memory and runs Ubuntu 18.04 OS, and the NVIDIA BlueField-2 DPU, which has 16 GB of memory and runs Ubuntu 20.04 OS. The development and profiling tools used in the experiments are DPDK (version 20.11.3.1.18) and DOCA (version 1.2.1). Each experiment is repeated more than three times, and the average values are reported for analysis.

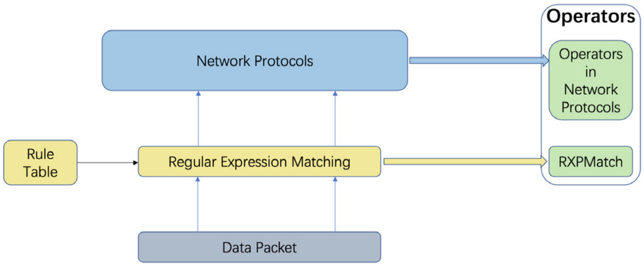


Fig. 7. The operators extracted in DPI.

4.2.2. Validate the representativeness, diversity and coverage of the operator set of DPUBench

To evaluate the representativeness, diversity, and coverage of the workload characteristics of Operator Set in DPUBench, we have selected 6 real workloads from the network, storage, and security scenarios for comparison. These selected workloads are representative application programs that are primarily offloaded by the DPU or essential components in real network applications. Each workload can be implemented using the combination of operators in DPUBench. The detailed information on these workloads, along with their corresponding operators, is presented in Table 5. This validation process also confirms the effectiveness of the two rules (Rule1 and Rule2) for extracting the representative operators, as discussed in Section 4.1.

The radar charts presented in Fig. 8 illustrate seven workload characteristics of the operators in DPUBench: IPC, iTLB-Miss-Ratio, dTLB-Miss-Ratio, L1D-Cache-Miss-Ratio, Integer instruction ratio, Branch instruction ratio, and Load&Store instruction ratio. The shapes of these radar charts demonstrate the diversity of workload characteristics covered by the Operator Set in DPUBench.

In Fig. 9, we compare the coverage of workload characteristics between the set of real workloads and the Operator Set in DPUBench. The radar charts representing the real workloads show that most of their workload characteristics can be effectively covered by the composed operators in DPUBench. This comparison validates the capability of the Operator Set in capturing the workload characteristics of real-world applications.

In addition to the radar charts, we have employed Principal Component Analysis (PCA) [35] to compare the diversity and coverage of workload characteristics between the Operator Set in DPUBench and the set of real workloads. The results are presented in Fig. 10. For visualization purposes, we have selected the top four principal components that contribute the most to the variance, accounting for 84.8% of the total variance contribution.

The area covered by the principal components of the Operator Set in DPUBench is capable of encompassing the area covered by the principal

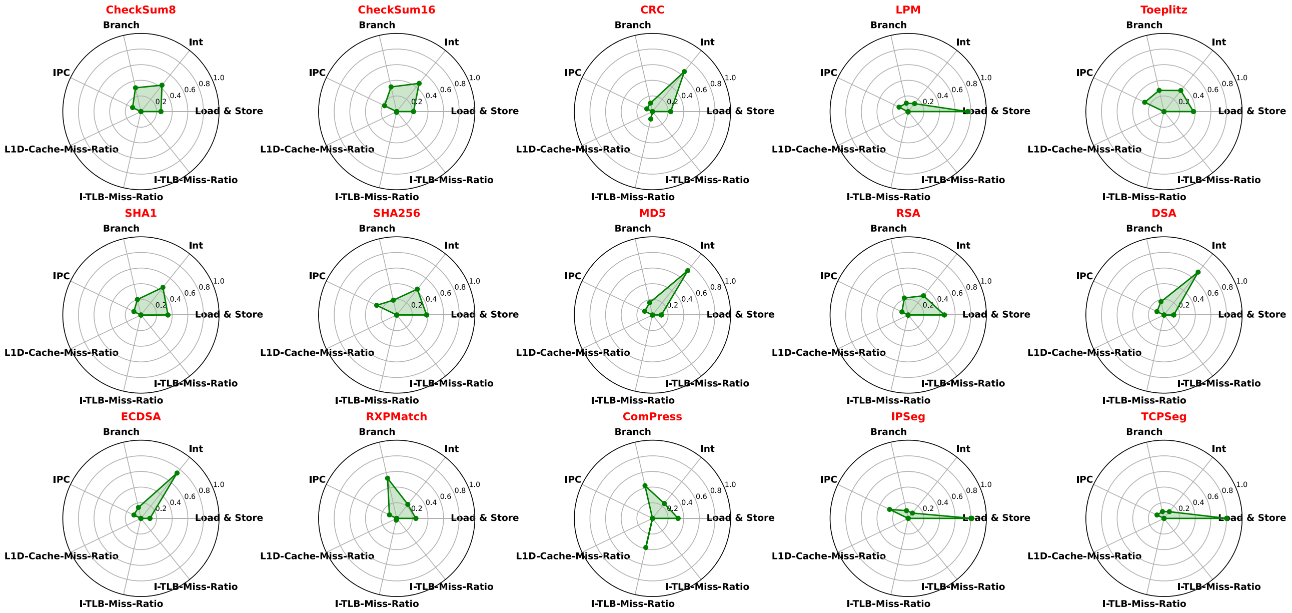


Fig. 8. The coverage experiment results of Operator Set under single thread 64B packet size configuration.

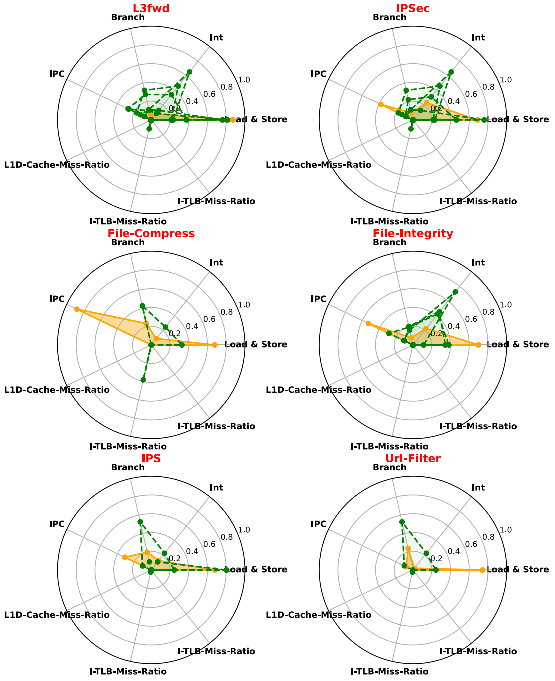


Fig. 9. Compare the coverage of operators with real application programs for validation. The composed operators for each real application program are shown in Table 5.

components of the real workloads, and it covers a significant portion of that area as well.

In conclusion, all of the experimental results consistently demonstrate that the Operator Set in DPUBench provides better coverage of workload characteristics compared to real workloads. Furthermore, the workload characteristics covered by the Operator Set in DPUBench exhibit a diverse range in terms of IPC, iTLB-Miss-Ratio, dTLB-Miss-Ratio, L1D-Cache-Miss-Ratio, Integer instruction ratio, Branch instruction ratio, and Load&Store instruction ratio.

4.2.3. Evaluate the NVIDIA BlueField-2 using operator set of DPUBench

We have conducted a performance evaluation on the NVIDIA BlueField-2 DPU, specifically the BlueField-2 model with encryption disabled and 25GbE capability. To ensure the micro-benchmarks, which are based on the operators in DPUBench, are representative, we have implemented them using general optimizations commonly used in real DPU applications. These optimizations include multi-threading, resource pooling, and cache-line alignment.

The micro-benchmarks can be configured with different input data sizes and number of threads. In our experiments, we have used input data sizes of 64B, 128B, 256B, 512B, and 1024B, which facilitates cache-line alignment. The number of threads can be configured as 1, 2, 4, 6, 8, 12, or 16. It is worth noting that the number of threads set to 16 exceeds the number of physical cores on the CPU (12 cores) and the number of ARM cores on the NVIDIA BlueField-2 (8 cores). The results of the performance evaluation are presented in Fig. 11. Each sub-figure correspond to the experiment result for one operator in the Operator Set, and each line in the figure shows micro-benchmark throughput for the different number of threads. Different lines in the same sub-graph correspond to different input data sizes.

The throughput of 10 operators, such as CRC, Checksum, toeplitz, RSA, DSA, ECDSA, AES, SHA256, SHA1, and MD5, exhibits linear scalability with the number of threads and is constrained by the number of physical cores. In these cases, the throughput increases proportionally with the number of threads, and the limiting factor is the number of available cores. Additionally, the throughput of these operators shows a fluctuation of approximately 10% to 20% when varying the input data size. This level of fluctuation is within the normal range, considering that the experiments are conducted under the same thread number and input data configuration, and the observed throughput variations fall within a consistent range.

The throughput of operators LPM, TCPSeg, and IPSeg demonstrates linear scaling with the input data size. This behavior can be attributed to the fact that these operators process a fixed amount of data within the input data. Consequently, in real-world applications, we can reduce the workload by encapsulating the data into fewer packets.

The Compress and RXPMatch operators are implemented using DOCA and deployed on the dedicated hardware accelerator of the BlueField-2. Their performance is not constrained by the number of physical cores available on the BlueField-2. Hence, when deploying these operators on the BlueField-2, it is possible to utilize more threads

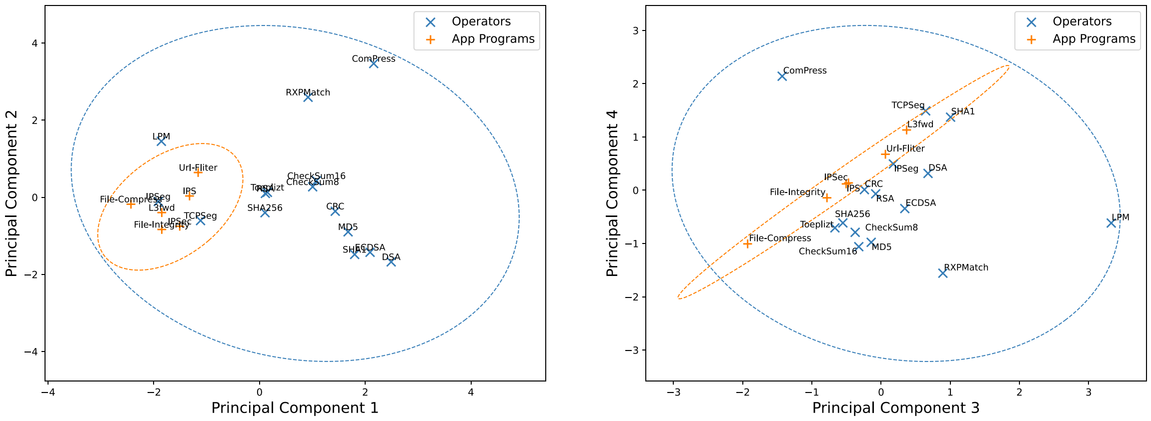


Fig. 10. PCA visualization results of Operator Set in DPUBench and typical Application Programs.

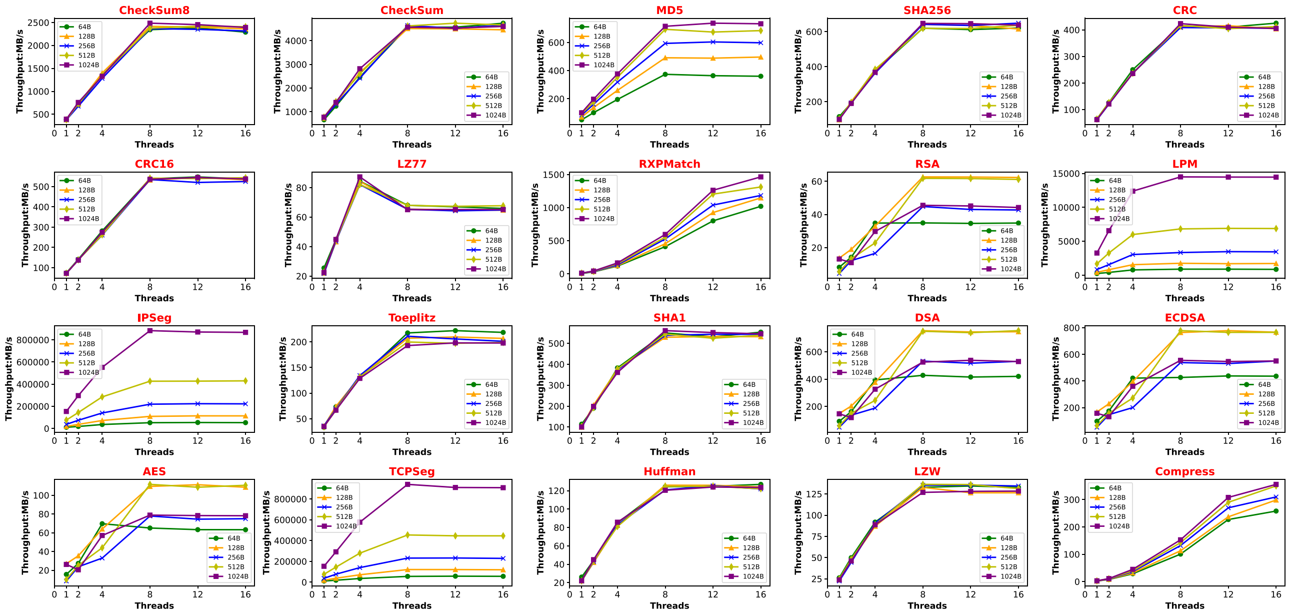


Fig. 11. The throughput of NVIDIA BlueField-2 using Operator Set in DPUBench under different data packet size and threads.

than the number of physical cores to further enhance their performance.

The comparison of throughput ratios between deploying micro-benchmarks on the NVIDIA BlueField-2 and Intel CPU is depicted in Fig. 12. For the majority of operators, the throughput is lower when deployed on the BlueField-2 compared to the Intel CPU. However, two exceptions are observed for the RXPMatch and Compress operators. These operators exhibit peak performance that can be 1.5 to 2 times higher when deployed on the BlueField-2 than on the Intel CPU. Consequently, applications such as IPS and Url-Filter (which utilize the RXPMatch operator) and NVMe-oF and File-Compress (which utilize the Compress operator) are more suitable for deployment on the BlueField-2 rather than the Intel CPU based on our experimental findings.

5. End-to-end evaluation programs of DPUBench

End-to-end Evaluation Programs are the other component of the solution instantiation in DPUBench, as mentioned in Section 3. In this section, we will outline the framework of End-to-end Evaluation Programs in DPUBench, which consists of both Client and Server. We will then provide the workloads of End-to-end Evaluation Programs in DPUBench and use them to do an evaluation of NVIDIA BlueField-2. Finally, We will present the experimental results of the End-to-end Evaluation Programs, which show the advantages of DPU in data centers.

5.1. The framework of end-to-end evaluation programs in DPUBench

Fig. 13 shows the framework of End-to-end Evaluation Programs in DPUBench, consisting of a Client with a dataset and traffic generator and a Server with applications that DPU can offload and cannot offload. Client device only contains CPU and is used for generating and sending data to the network. The server device can be a node that only contains a CPU or a node contains both CPU and DPU in data centers. The server receives and processes data packets sent by the Client, and transmits the results to the Client through the network.

The dataset in Client is used to generate the data and the traffic generator is used to send data packets with specified traffic according to network protocols. The network applications in Server are the application programs that DPU can offload and implement by DOCA SDK and DPDK SDK for DPU and programs for CPU. The applications in

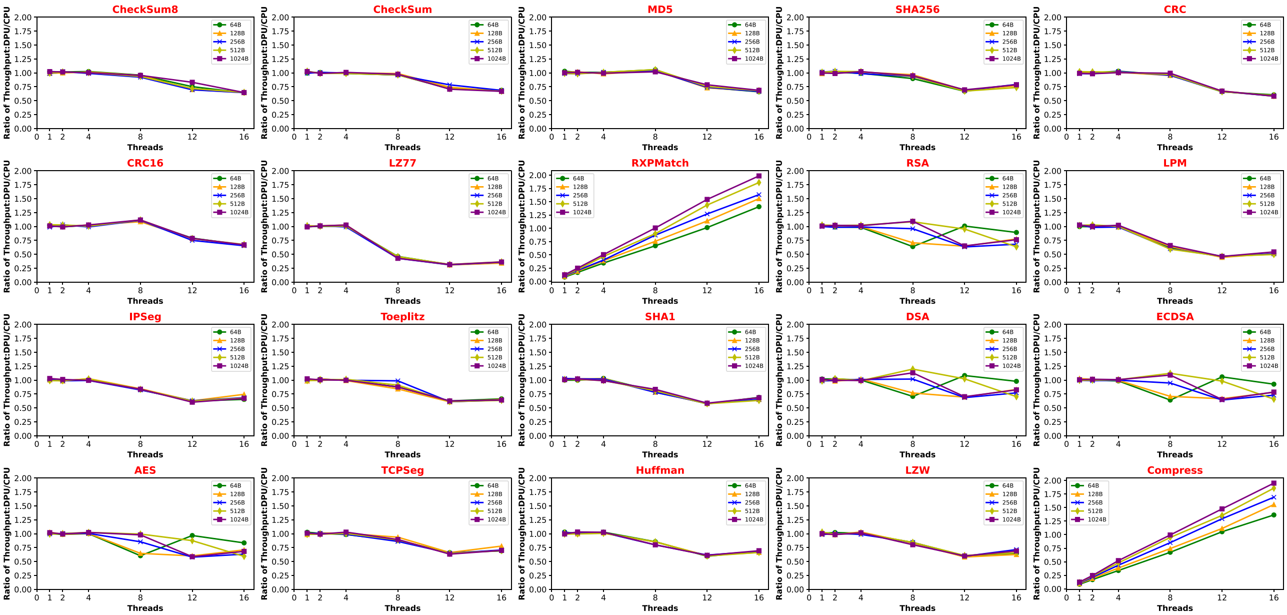


Fig. 12. The ratio of throughput of NVIDIA BlueField-2 and Intel CPU using Operator Set in DPUBench under different data packet sizes and threads.

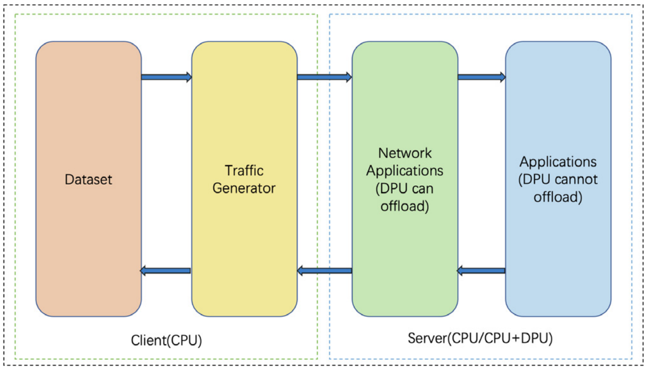

Dataset
Traffic Generator
Network Applications (DPU can offload)
Applications (DPU cannot offload)
Client(CPU)
Server(CPU/CPU+DPU)

Fig. 13. The framework of end-to-end evaluation programs in DPUBench.

the Server are application programs that DPU cannot offload and are executed by the CPU.

5.1.1. *The workloads*

We have implemented two end-to-end DPU workloads in DPUBench. The first workload focuses on evaluating the performance of offloading the flow table to the DPU. This process is crucial as it enables the implementation of various network applications such as Packet Filters, Quality of Service, and Load Balancing. However, it should be noted that the first workload does not encompass the complete packet processing procedure found in real-world applications.

To address this limitation, we have developed a second end-to-end DPU workload that closely resembles a real application structure. By evaluating this workload, we gain deeper insights and uncover additional information that may remain hidden when conducting experiments solely on the first workload.

5.2. *The experiments of end-to-end evaluation programs of DPUBench*

The structure of the two end-to-end DPU workloads is based on Fig. 13. In the first workload, we utilize pktgen, which is available in the released version of the Linux kernel, as the traffic generator. And the dataset consists of randomly generated data by pktgen. The logic of the network application is straightforward: when packets arrive at the Server node, the Server searches the entries in the flow table based on the 5-tuple information (source IP, destination IP, destination port IP, source MAC, destination MAC) extracted from the packet header. If the tuple matches an existing entry, the count value of that entry is incremented. Otherwise, a new entry is added to the flow table. It is important to note that in this workload, the flow table is typically populated solely from the application layer, and therefore, the workload does not encompass the application part illustrated in Fig. 13.

The second workload encompasses both the network application and application parts depicted in Fig. 13. The network application in this workload focuses on application recognition, which involves inspecting the payload of received packets to determine if they contain specific character strings based on regular expression matches. Depending on the recognition results, the application performs different tasks. These tasks include calculating the hash value of the entire packet, compressing the payload section of the packet, or directly sending the packet back to the Client. By incorporating both the network application and application parts, this workload can provide a complete packet processing procedure compared to the first workload.

5.2.1. *Experimental configurations*

The Server is the same as the configurations mentioned in Section 4.2.1 and we just introduce the Client. We deploy the experiments on the Intel Xeon E5-2620 v3 CPU (4 processors) equipped with 10 GB of memory for the Client. The OS is Ubuntu 20.04 and the profiling tool is DPDK (version 20.11.3.1.18). We also repeat each experiment more than three times and report the average values.

5.2.2. *Evaluate the NVIDIA BlueField-2 using end-to-end evaluation programs of DPUBench*

In Fig. 14, the throughput and packet loss ratio are depicted for different time intervals between packets sent by the Client. The network application is accomplished in two ways: hardware and software. The hardware version implementation uses BlueField's flow table offloading capacity and all of the flow table operations are offloaded to BlueField, while in the software implemented version, the flow table and

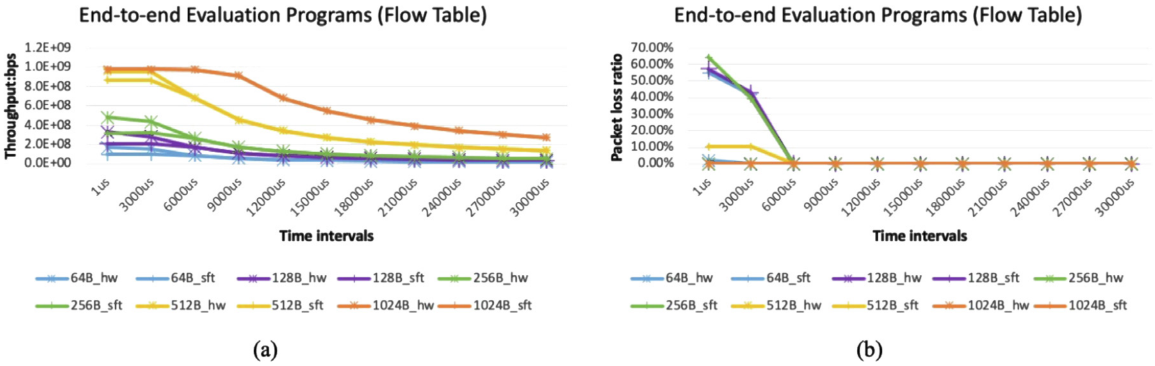


Fig. 14. The throughput and packet loss ratio of NVIDIA BlueField-2 using the first end-to-end DPU workload (flow-table) in DPUBench under different data packet sizes and time intervals. Fig. 14(a) corresponds to experiment results for throughput, Fig. 14(b) shows experimental results for packet loss rate. 64B_sft means software-implemented flow table under 64 Byte packet size, and 64B_hw means hardware-implemented flow table under 64B packet size.

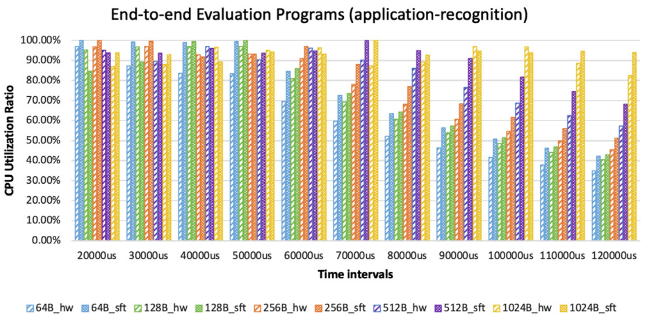


Fig. 15. The Server CPU utilization ratio for the second end-to-end DPU workload (application-recognition) in DPUBench under different data packet size and time intervals.

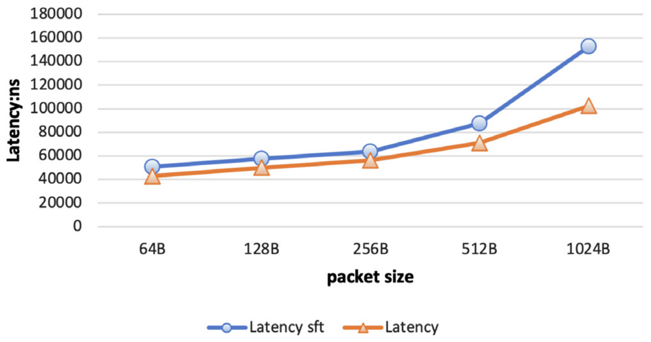


Fig. 16. The latency of the second end-to-end DPU workload (application recognition) in DPUBench under different data packet size.

all of the flow table operations are implemented by the C++ codes. For ease of comparison, the results for both software and hardware implementations are presented in the same sub-figure. Additionally, the experiments are conducted with different packet sizes sent by the Client, and a different color in the sub-figure represents each packet size.

The experiment results demonstrate that offloading the flow table to the DPU can yield greater throughput improvement when the packet size sent by the Client is small and when the Client sends packets at a fast speed (with low time intervals). This can be attributed to the fact that with larger packets, the Client's ability to send packets at maximum speed is limited by the network port's bandwidth. In such cases, the data processing speed does not become a bottleneck, and therefore, the DPU's ability to improve the system's throughput is limited. Conversely, when the Client sends packets at high speed that exceeds the Server's processing capacity, packet loss occurs. Offloading the flow table to the DPU can provide greater throughput improvement and reduce the packet loss ratio in such situations since the DPU can process packets at a faster speed.

Fig. 15 illustrates the Server CPU utilization ratio when the Client sends packets of different packet sizes at varying time intervals. For ease of comparison, the experiment results are presented in two ways: offloading the application recognition to the DPU (NVIDIA BlueField-2) and deploying the application on the Server CPU, depicted in the same sub-figure. The results indicate that when the Client sends packets at a slow speed, although offloading the application to the DPU does not improve the system's throughput, it can reduce the Server CPU utilization ratio by 10% to 20%.

Fig. 16 presents the results of experiments conducted to measure the reduction in process delay achieved by deploying the app-recognition workload on NVIDIA BlueField-2. The results indicate that offloading the application recognition to the DPU can result in a decrease of 10% to 15% in process delay. However, this reduction ratio is lower compared to the results obtained for the RXPMatch operator. This is because the packet processing procedure in the app-recognition workload involves both processing on the Server CPU and on the DPU (NVIDIA BlueField-2), and the DPU can only accelerate the latter part of the processing procedure.

6. Related work

Modern computer chips can be broadly categorized into two types: general-purpose processors (CPU) and specialized processors designed for specific acceleration tasks, including GPU, TPU [4], and DPU. As the performance gains predicted by Moore's Law [36] and Dennard's Scaling Law [3] have been slowing down, CPU performance improvement is also slowing down. This poses a challenge in meeting the increasing demand for computing resources in emerging fields like artificial intelligence, big data, and the Internet of Things. To address this challenge, specialized processors optimized for specific acceleration tasks are being developed. For instance, as the field of artificial intelligence has grown, AI models have significantly increased in size and complexity, with parameters ranging from 62 million in AlexNet [37] to 175 billion in GPT-3 [38] and beyond, with even larger models on the horizon.

As chip development progresses, research on evaluating various types of chips is also ongoing, with benchmarks being a key focus. CPU evaluation is supported by representative benchmarks such as SPEC CPU [39] provided by the Standard Performance Evaluation

Corporation (SPEC), which assesses single-core performance, and PARSEC [40] provided by Princeton University, which evaluates multi-core performance. The latest version of SPEC CPU is SPEC CPU2017 [41], which is the sixth iteration of the benchmark. Additionally, a new version of SPEC CPU, temporarily referred to as SPEC CPU v8 [42], is currently in development.

For evaluating AI chips, there are representative benchmarks such as MLPerf [43] and AIBench [44]. MLPerf focuses on selecting models from various AI tasks, including image classification, object detection, and machine translation, and constructs workloads for both training and inference evaluations [43]. AIBench, on the other hand, extracts 13 operators from typical AI scenarios and constructs micro-benchmarks using these operators [44]. These benchmarks provide standardized and comprehensive evaluation metrics for assessing the performance of AI chips in different AI tasks.

In the field of DPU evaluation, there are several existing studies and benchmarks. NVIDIA has developed RXPBench [12] using the DOCA SDK for their BlueField DPU, which currently focuses on regular expression matching as the evaluation program and measures the execution time on the chip. Amazon has conducted performance improvement tests in virtual machines by deploying the hypervisor on Nitro chips [13]. YUSUR has developed evaluation programs for their four DPU products [7–10] in different scenarios, such as using SQL queries for financial scenarios and measuring query latency [8]. Wei et al. [14] and Sun et al. [45] have evaluated the latency and throughput of NVIDIA BlueField-2 and provided optimization recommendations for DPUs with specific characteristics. These studies contribute to assessing and optimizing DPUs in specific architectures and scenarios.

As a new generation of programmable SmartNIC, the evaluation studies on SmartNIC can provide valuable insights for designing DPU benchmarks. Ibanez et al. [15] introduced the wire-to-wire latency metric to measure the time taken from receiving RPC requests to sending them over the network, using a SmartNIC placed in the network. Ma et al. [16] evaluated the performance of matrix multiplication and other operators in AI applications on a SmartNIC for distributed AI training, as well as the impact on AI model training time after deploying the SmartNIC in a distributed training framework. Mandal et al. [17] evaluated a SmartNIC for storage applications, focusing on the throughput of processing storage system read and write requests after connecting the SmartNIC to the network. Sabin et al. [18] evaluated a SmartNIC for security applications, measuring the throughput of processing encrypted communication requests in the corresponding scenarios. Bosshart et al. [19] offloaded a network layer protocol using a SmartNIC for SDN and evaluated the latency of communication with a data center node where the SmartNIC was deployed. These studies provide valuable performance metrics and insights that can inform the design of DPUBench.

7. Conclusion and plan

In conclusion, we have proposed DPUBench, an application-driven scalable benchmark suite for comprehensive DPU evaluation. DPUBench follows a methodology comprising problem definition, problem instantiation, and solution instantiation. We focus on network applications and select network, storage, and security as typical application scenarios. We extract essential operators from these scenarios and develop end-to-end evaluation programs, forming the Operator Set and Workload Programs of DPUBench. We present evaluation results of the NVIDIA BlueField-2 using DPUBench and provide optimization recommendations. DPUBench will be continuously maintained and updated to keep pace with DPU's development, and we will evaluate other DPUs in our future version of DPUBench. We will also investigate the IO virtualization application scenario in the future, as it plays a vital role in modern data centers.

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] J. Zhan, A BenchCouncil view on benchmarking emerging and future computing, BenchCouncil Trans. Benchmarks, Stand. Eval. (2022) 100064.
[2] J. Shalf, The future of computing beyond Moore's law, Phil. Trans. R. Soc. A 378 (2166) (2020) 20190061.
[3] R.H. Dennard, F.H. Gaensslen, H.-N. Yu, V.L. Rideout, E. Bassous, A.R. LeBlanc, Design of ion-implanted MOSFET's with very small physical dimensions, IEEE J. Solid-State Circuits 9 (5) (1974) 256–268.
[4] N.P. Jouppi, C. Young, N. Patil, D. Patterson, G. Agrawal, R. Bajwa, S. Bates, S. Bhatia, N. Boden, A. Borchers, et al., In-datacenter performance analysis of a tensor processing unit, in: Proceedings of the 44th Annual International Symposium on Computer Architecture, 2017, pp. 1–12.
[5] The NVIDIA's definiton of DPU, https://resources.nvidia.com/en-us-accelerated-networking-resource-library/whats-a-dpu-data-product?lx=LbHvpR&topic=networking-cloud.
[6] NVIDIA BlueField-2, https://resources.nvidia.com/en-us-accelerated-networking-resource-library/bluefield-2-dpu-datasheet?lx=LbHvpR&topic=networking-cloud.
[7] YUSUR's DPU evaluation programs for cloud data center, https://www.yusur.tech/solution/cloudDataCenter.
[8] YUSUR's DPU evaluation programs for financial data calculation acceleration, http://www.yusur.tech/solution/financialDataCalculationAcceleration.
[9] YUSUR's DPU evaluation programs for high performance computing, https://www.yusur.tech/solution/highPerformenceComputing.
[10] YUSUR's DPU evaluation programs for industrial Internet, https://www.yusur.tech/solution/industrialInternet.
[11] The information of Intel Mount Evans, https://www.intel.com/content/www/us/en/newsroom/resources/press-kit-architecture-day-2021.html#gs.xbri9l.
[12] Doca document v1.5.1 :nvidia doca rxpbench user guide, https://docs.nvidia.com/doca/sdk/rxpbench/index.html.
[13] A. Liguori, The nitro project–next generation AWS infrastructure, in: Hot Chips: A Symposium on High Performance Chips, 2018.
[14] X. Wei, R. Chen, Y. Yang, R. Chen, H. Chen, A comprehensive study on off-path SmartNIC, 2022, arXiv preprint arXiv:2212.07868.
[15] S. Ibanez, A. Mallery, S. Arslan, T. Jepsen, M. Shahbaz, N. McKeown, C. Kim, The nanoPU: Redesigning the CPU-network interface to minimize RPC tail latency, 2020, arXiv preprint arXiv:2010.12114.
[16] R. Ma, E. Georganas, A. Heinecke, S. Gribok, A. Boutros, E. Nurvitadhi, FPGA-based AI smart NICs for scalable distributed AI training systems, IEEE Comput. Archit. Lett. 21 (2) (2022) 49–52.
[17] P.C. Mandal, N. Mariyappa, S. Das, A. Venkataraman, Storage Offload on SmartNICs.
[18] G. Sabin, M. Rashti, Security offload using the SmartNIC, A programmable 10 Gbps ethernet NIC, in: 2015 National Aerospace and Electronics Conference, NAECON, IEEE, 2015, pp. 273–276.
[19] P. Bosshart, G. Gibb, H.-S. Kim, G. Varghese, N. McKeown, M. Izzard, F. Mujica, M. Horowitz, Forwarding metamorphosis: Fast programmable match-action processing in hardware for SDN, ACM SIGCOMM Comput. Commun. Rev. 43 (4) (2013) 99–110.
[20] R. Recio, B. Metzler, P. Culley, J. Hilland, D. Garcia, A remote direct memory access protocol specification, Technical Report RFC 5040, October, 2007.
[21] G.F. Pfister, An introduction to the infiniband architecture, in: High Performance Mass Storage and Parallel I/O, Vol. 42, (617–632) 2001, p. 102.
[22] NVIDIA BlueField-3, https://resources.nvidia.com/en-us-accelerated-networking-resource-library/datasheet-nvidia-bluefield?lx=LbHvpR&topic=networking-cloud.
[23] G.R. Wright, W.R. Stevens, TCP/IP Illustrated, Volume 2 (Paperback): The Implementation, Addison-Wesley Professional, 1995.
[24] N. Doraswamy, D. Harkins, IPSec: The New Security Standard for the Internet, Intranets, and Virtual Private Networks, Prentice Hall Professional, 2003.
[25] C. Guo, H. Wu, Z. Deng, G. Soni, J. Ye, J. Padhye, M. Lipshteyn, RDMA over commodity ethernet at scale, in: Proceedings of the 2016 ACM SIGCOMM Conference, 2016, pp. 202–215.
[26] B. Pfaff, J. Pettit, T. Koponen, E. Jackson, A. Zhou, J. Rajahalme, J. Gross, A. Wang, J. Stringer, P. Shelar, et al., The design and implementation of open vswitch, in: 12th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 15), 2015, pp. 117–130.
[27] R. Russell, Virtio: towards a de-facto standard for virtual I/O devices, Oper. Syst. Rev. 42 (5) (2008) 95–103.
[28] D. Minturn, Nvm express over fabrics, in: 11th Annual OpenFabrics International OFS Developers' Workshop, 2015.
[29] OpenSSL, https://www.openssl.org.

[30] J. Ziv, A. Lempel, A universal algorithm for sequential data compression, IEEE Trans. Inform. Theory 23 (3) (1977) 337–343.

[31] D.A. Huffman, A method for the construction of minimum-redundancy codes, Proc. IRE 40 (9) (1952) 1098–1101.

[32] W. Diffie, M.E. Hellman, New directions in cryptography, in: Democratizing Cryptography: The Work of Whitfield Diffie and Martin Hellman, 2022, pp. 365–390.

[33] D. Joan, R. Vincent, The design of Rijndael: AES-the advanced encryption standard, Inf. Secur. Cryptogr. (2002).

[34] D.B. Johnson, A.J. Menezes, Elliptic curve DSA (ECDSA): an enhanced DSA, in: Proceedings of the 7th Conference on USENIX Security Symposium, Vol. 7, 1998, pp. 13–23.

[35] I.T. Jolliffe, Principal Component Analysis for Special Types of Data, Springer, 2002.

[36] G.E. Moore, et al., Cramming More Components onto Integrated Circuits, McGraw-Hill New York, 1965.

[37] A. Krizhevsky, I. Sutskever, G. Hinton, Imagenet classification with deep convolutional networks, in: Proceedings of the 26th Annual Conference on Neural Information Processing Systems, NIPS, pp. 1106–1114.

[38] 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.

[39] SPEC CPU, https://www.spec.org./benchmarks.html#cpu.

[40] PARSEC, https://parsec.cs.princeton.edu/index.htm.

[41] 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.

[42] SPEC CPU v8, https://www.spec.org/cpuv8.

[43] 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.

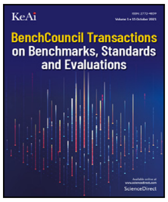
[44] W. Gao, F. Tang, J. Zhan, X. Wen, L. Wang, Z. Cao, C. Lan, C. Luo, X. Liu, Z. Jiang, Aibench scenario: Scenario-distilling ai benchmarking, in: 2021 30th International Conference on Parallel Architectures and Compilation Techniques, PACT, IEEE, 2021, pp. 142–158.

[45] S. Sun, C. Huang, R. Zhang, L. Chen, Y. Huang, M. Yan, J. Wu, A comprehensive study on optimizing systems with data processing units, 2023, arXiv preprint arXiv:2301.06070.

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

StreamAD: A cloud platform metrics-oriented benchmark for unsupervised online anomaly detection

Jiahui Xu [a,b,1], Chengxiang Lin [a,b,1], Fengrui Liu [a,b,1], Yang Wang [a,b], Wei Xiong [a,b], Zhenyu Li [a,b], Hongtao Guan [a,b], Gaogang Xie [b,c,*]

[a] *Institute of Computing Technology, Chinese Academy of Sciences, China*
[b] *University of Chinese Academy of Sciences, China*
[c] *Computer Network Information Center, Chinese Academy of Sciences, China*

ARTICLE INFO

Keywords:
Cloud metrics
Online machine learning
Anomaly detection
Streaming data
Time series

ABSTRACT

Cloud platforms, serving as fundamental infrastructure, play a significant role in developing modern applications. In recent years, there has been growing interest among researchers in utilizing machine learning algorithms to rapidly detect and diagnose faults within complex cloud platforms, aiming to improve the quality of service and optimize system performance. There is a need for online anomaly detection on cloud platform metrics to provide timely fault alerts. To assist Site Reliability Engineers (SREs) in selecting suitable anomaly detection algorithms based on specific use cases, we introduce a benchmark called StreamAD. This benchmark offers three-fold contributions: (1) it encompasses eleven unsupervised algorithms with open-source code; (2) it abstracts various common operators for online anomaly detection which enhances the efficiency of algorithm development; (3) it provides extensive comparisons of various algorithms using different evaluation methods; With StreamAD, researchers can efficiently conduct comprehensive evaluations for new algorithms, which can further facilitate research in this area. The code of StreamAD is published at https://github.com/Fengrui-Liu/StreamAD.

1. Introduction

Cloud platform [1] is a type of computing infrastructure that provides hardware and software resources over the internet, such as virtual machines, storage, and networking capabilities. It can facilitate the developers building and deploying software applications.

With the growing market of cloud platform, its scale has become enormous. However, the prosperity of cloud platforms also brings significant challenges to Site Reliability Engineers (SREs) in detecting and diagnosing faults within large-scale cloud platforms. The computing infrastructures providing services need to guarantee Service Level Agreements (SLAs) to customers. Unexpected service downtime can greatly impact stability objectives and lead to substantial financial losses.

Benefiting from the intuitive visualization form of time-series metric data, such as metric dashboards, they are often the primary objects for anomaly detection in cloud platforms. SREs can easily point out whether the collected metrics are as expected. In order to reduce labor and enhance the quality of service, major cloud providers such as Microsoft Azure [2], Google Cloud [3], Amazon Cloud [4] and Alibaba Cloud [5] have adopted machine learning and artificial intelligence technologies to assist SREs in detecting anomalies.

Metrics anomaly detection presents a challenging task due to the following reasons [6,7]:

- Lack of labeled data. Anomalous data is rare compared to normal data, and identifying specific anomalies that warrant attention can be difficult. Practical application scenarios are open-ended, making it difficult to define the anomalies that should be detected. The confirmation of specific anomalies, such as their beginning and duration, requires reliable input from SREs. As a result, obtaining accurate labels is challenging [8]. The manually labeling process is also prone to errors [7], with a wide-ranging discussions regarding the flaws in current public datasets. This issue stems from the subjective judgments made while assigning ground truth labels. The lack of labeled data presents challenges in designing, training, and evaluating models effectively.
- Online detection. Cloud platform metrics are often monitored in real-time, in order to quickly alert when a fault is detected. This helps reduce the mean time to repair (MTTR), which is

[*] Corresponding author.
E-mail addresses: xujiahui21b@ict.ac.cn (J. Xu), linchengxiang21s@ict.ac.cn (C. Lin), liufengrui18z@ict.ac.cn (F. Liu), wangyang2013@ict.ac.cn (Y. Wang), xiongwei20b@ict.ac.cn (W. Xiong), zyli@ict.ac.cn (Z. Li), guanhongtao@ict.ac.cn (H. Guan), xie@cnic.com (G. Xie).
[1] These authors contributed equally to this work.

https://doi.org/10.1016/j.tbench.2023.100121
Received 3 May 2023; Received in revised form 3 July 2023; Accepted 5 July 2023
Available online 6 July 2023

critical for maintaining service level agreements. Thus, there is a high demand for algorithms to accurately and efficiently detect metric anomalies with an online manner. An effective online anomaly detection algorithm must continually process incoming data streams and update its model online to ensure accurate and reliable detection. In the situations when metrics experience significant changes in data distribution, known as concept drift [9], algorithms need to adapt to these changes promptly to prevent false alarms from occurring.

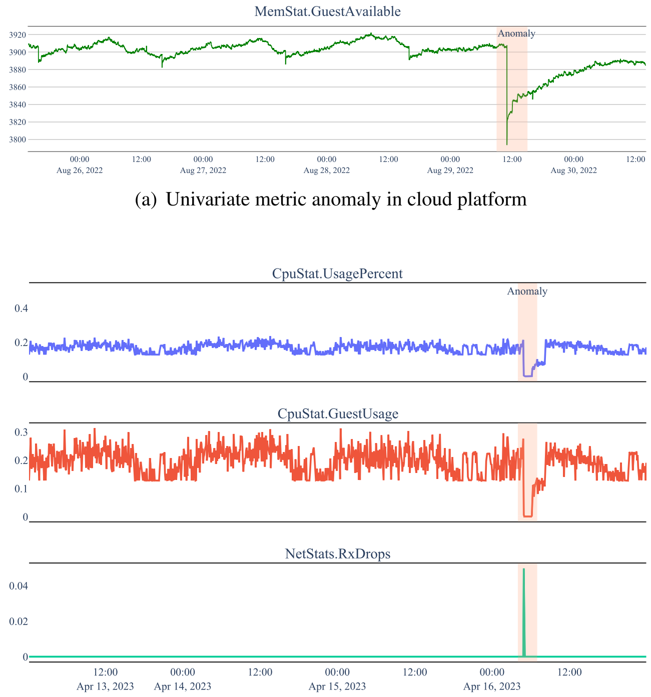
- Data dimension. A cloud platform metric can describe a specific aspect of a cloud platform, such as CPU utilization, network received packets or memory usage. Each metric is represented as an univariate time series, where the series is independent of others, i.e. the data dimension is univariate. However, in the event of a host fault in a cloud platform, multiple metrics may exhibit anomalous behavior. An underlying assumption is that there is internal interaction between different metrics. Thus, they can be used together to detect anomalies utilizing a process known as multivariate detection. However, simply combining the anomaly detection results of each univariate time series performs poorly for multivariate anomaly detection methods [10]. This naive approach fails to capture the inter-dependencies among metrics within a service. Therefore, there is a growing need for dedicated algorithms that can effectively handle multivariate data streams.
- Domain-specific datasets and benchmarks. Although researchers have published several datasets and benchmarks for anomaly detection [11–13], they are not specific to a particular domain. Nevertheless, there are significant differences in data characteristics across various fields. For instance, ECG datasets [14], voice datasets [15], and cloud platform metrics datasets [8] are all in time-series format, they can differ greatly in periodicity, range of values, and other key characteristics. The lack of domain-specific datasets and benchmarks for cloud platform metrics still persists.

As can be seen from the aforementioned challenges, metrics anomaly detection algorithms for real cloud platforms need to be unsupervised, since high-quality labeled data may not always be accessible. Additionally, these algorithms should detect anomalies with an online manner, enabling them to report fault alarms timely.

Although researchers have designed and contributed various unsupervised algorithms for online anomaly detection, there is a lack of a comprehensive benchmark to evaluate their effectiveness in cloud platform metrics. To tackle above issue, we propose StreamAD, which is a domain-specific benchmark for unsupervised online anomaly detection of cloud platform metrics. The primary contributions of StreamAD are summarized as follows:

- StreamAD collects eleven unsupervised online anomaly detection algorithms, encapsulating them using a unified and easy-to-use application programming interface (API). It can serve as an out-of-the-box anomaly detection module for quick case validation. All the code is open-source.
- We abstracts various common operators for different online anomaly detection, accompanied by data process methods. It can greatly facilitate researchers using StreamAD to develop new algorithms.
- StreamAD focuses on cloud platform metrics dataset, providing extensive comparisons of various algorithms using different evaluation methods. The results form a benchmark for cloud platform metrics anomaly detection.

StreamAD is dedicated to quickly verifying the effectiveness of different algorithms on use cases, enabling the application of machine learning-based algorithms for real-world cloud platform metrics. It also helps researchers in rapidly developing and comparing new algorithms, promoting further research and development in this rapidly evolving research domain.



(a) Univariate metric anomaly in cloud platform

(b) Multivariate metrics anomaly in cloud platform

Fig. 1. Example of anomalies for cloud platform metrics.

2. Background

2.1. Cloud platform metrics

The primary focus of anomaly detection is on the observable data objects within cloud platforms. Observability refers to the ability to monitor and comprehend the operational state of a system's underlying infrastructure, platform, and applications through their external outputs. In a complex cloud platform system, observability assists in describing the system's current status, verifying the proper execution of each component's intended logic, identifying performance bottlenecks, and tracking optimizations for better system management. In the event of anomalies, observable data objects play a crucial role in real-time data collection and visualization of various key metrics. By analyzing these observable data, SREs can swiftly identify and address faults within complex cloud platforms, leading to optimized system performance and enhanced system reliability.

Metrics serve as a fundamental component of observable data objects in cloud platforms. A metric is a numerical value or counter that represents the state of the system, which is atomic and cumulative. Each metric can be regarded as a logical measurement unit, typically representing data statistics updated over time. Although the specific metrics monitored by vary cloud platforms can be different. Take Google Cloud metrics [16] as an example. A typical set of cloud platform metrics includes five categories, including CPU, System, Memory, Block, and Network, which can cover various aspects of the cloud platform.

As each cloud-platform metric can be represented in a time series, it is natural to conduct independent analysis on each metric, namely univariate metric anomaly detection. For instance, a *MemStat.GuestAvaliable* metric from a cloud platform, as shown in Fig. 1(a). The metric is represented as time-series data which is continuously extended as long as it is under continuous observation. This nearly real-time data observation process enables the system status to be monitored online, making it possible to alert the faults in a timely manner.

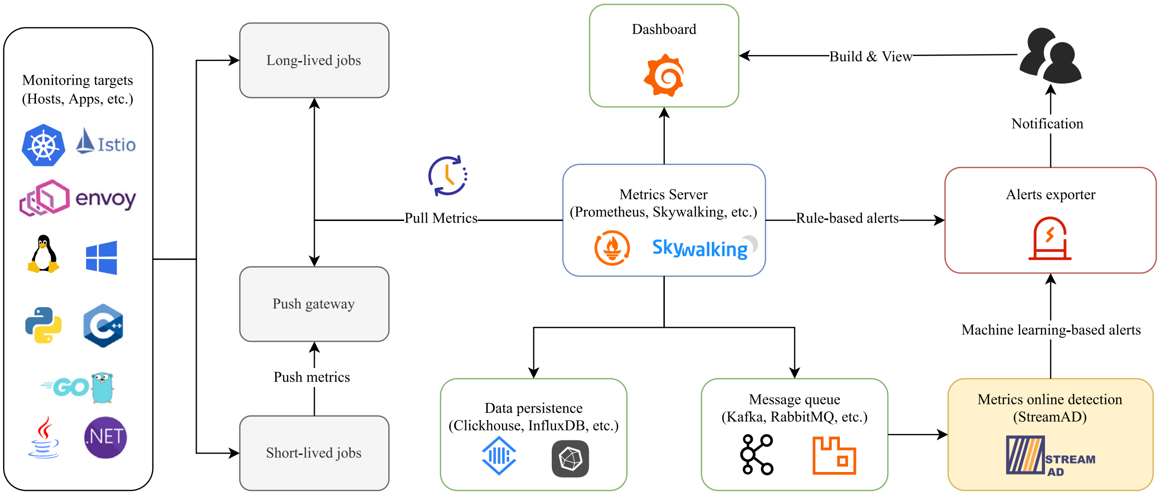


Fig. 2. Architecture example of metrics online detection in a cloud platform.

In addition, some cloud platform faults may be reflected in multiple metrics. Take Fig. 1(b) as an example, a network issue causes a sharp increase in the *NetStats.RxDrops* metric, accompanied with a decrease in the *CpuStat.UsagePercent* and *CpuStat.GuestUsage* metrics. The root cause of this fault is that the host has suffered from receiving packets (*RxPackets*) loss. The deployed services fail to response external requests, resulting in low CPU usage. In this case, multiple metrics exhibit abnormal behaviors during the fault. Under the assumption that there is an internal interaction among different metrics, they can be analyzed together, leading to achieve multivariate metrics anomaly detection. StreamAD covers for both univariate and multivariate metrics detection.

2.2. The role of metric anomaly detection

In real production environments of cloud platforms, as depicted in Fig. 2, different metrics can be collected and reformatted by various agents or probes. These metrics may include data from fundamental host machines, resource management controllers, and applications constructed using different programming languages. Both short-lived and long-lived jobs generate metrics in an online manner and export them to the metrics server using push and pull methods respectively. The metrics server stores data streams into a time series database, achieving data persistence that can be utilized for data retrieval and backtracking. Some popular metrics servers, such as Prometheus [17] and Skywalking [18], have built-in rule modules that allow SREs to implement anomaly detection by pre-setting rules. However, manual operations by setting alert rules struggle to adapt to large-scale cloud platforms, as they heavily rely on expert knowledge and are error-prone. Thus, fully-automated operation pipelines powered by machine learning capabilities become a promising approach for achieving SLA goals.

StreamAD can serve as a logical unit that is dedicated to utilizing machine learning technology for metric monitoring. It is capable of subscribing to message queues, allowing it to receive and analyze streaming data based on algorithmic processing logic. Once the observed data has been analyzed, it will be scored accordingly. Those data with a high anomaly score are then sent to the alert exporter, and then further notify the users in time. SREs can trace the metrics records via a customized dashboard and deal with the faults in the cloud platform.

3. Related work

Anomaly detection is a broad topic that has been applied in different applications, leading to significant research efforts over the years [24–27]. Researchers have devoted substantial effort publishing benchmarks, and we provide a summary of related work in Table 1.

ADBench [11] is a comprehensive anomaly detection benchmark that includes unsupervised, semi-supervised, and fully-supervised algorithms. It analyzes the performance of thirty algorithms under different types of anomalies by simulating different environments. However, this benchmark only focuses on tabular data, which may not be suitable for time-series data in cloud platforms.

TODS [13,19] constructs a full-stack automated machine learning system for anomaly detection. It is a benchmark that identifies four multivariate real-world datasets from different domains and benchmarks nine algorithms on synthetic and real-world datasets. TODS also publishes preprocess and synthetic scripts, as well as algorithm implementations.

NAB [20,21] focuses on scenarios of online anomaly detection in practical applications. Although all algorithms in this benchmark are designed for online anomaly detection and use a scoring algorithm designed for streaming data, it only evaluates univariate anomaly detection algorithms and lacks the discussion of multivariate time series data. Furthermore, this benchmark is not currently maintained and does not cover new online anomaly detection algorithms.

Exathlon [22] is a benchmark that focuses on time series data. It provides a new analytical perspective on time series anomaly detection, which is the interpretability of the detection results. It focuses on Spark application monitoring and provides an end-to-end pipeline for explainable time series anomaly detection. However, this benchmark is for offline analysis.

UTSD [23] is a benchmark that focuses on univariate time series and provides a user-friendly visual interface for those series. This benchmark contributes a large number of datasets and their variants. However, the use case of this benchmark directly applies tabular data anomaly detection algorithms to time series data, which has great limitation on modeling the features of time series data.

TSB-UAD [12] is benchmark for univariate time series. It contributes a principled methodology for generating labeled anomaly detection datasets. It also reviews factors affecting the performance of methods. However, this benchmark is also for offline anomaly detection and cannot meet the requirements of online anomaly detection for cloud platforms.

Regarding the benchmark for anomaly detection of cloud platform metrics, it should have the following properties. Firstly, the benchmark should focus on time-series data, including both univariate and multivariate metrics. Secondly, the anomaly detection methods can be updated in a streaming manner, without periodic offline training. Finally, it requires an extensive validation on cloud platform metrics.

Table 1
Comparison of anomaly detection benchmarks across various properties.

Properties/Benchmark	# Algorithms	Time series	Multivariate	Streaming updates	Domain specific
ADBench [11]	30	✗	✓	✗	✗
TODS [13,19]	9	✓	✓	✗	✗
NAB [20,21]	12	✓	✗	✓	✗
Exathlon [22]	3	✓	✓	✗	✓
UTSD [23]	3	✓	✗	✗	✗
TSB-UAD [12]	12	✓	✗	✗	✗
StreamAD (Ours)	11	✓	✓	✓	✓

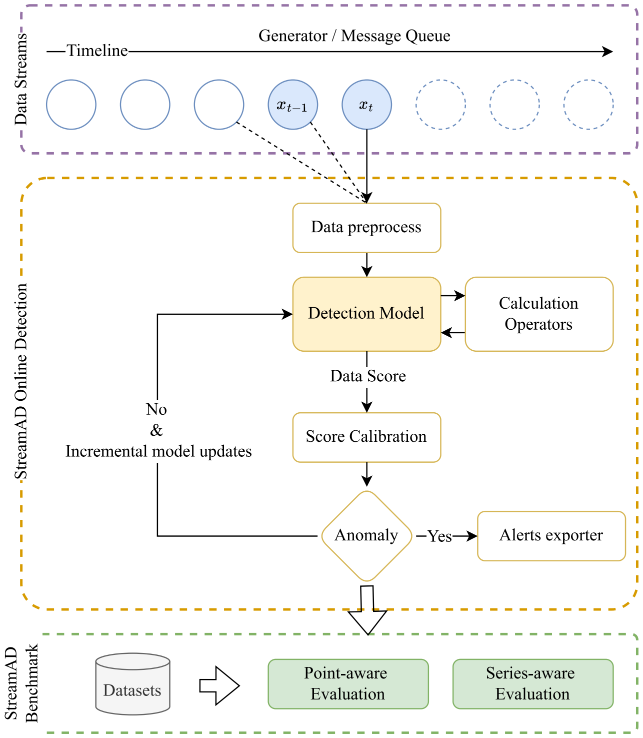


Fig. 3. The framework of StreamAD.

Table 2
Anomaly detection algorithms included in StreamAD.

	Algorithm	Sliding window	Seasonal
Univariate	KNN-CAD [31]	✓	✗
	SPOT [32]	✗	✗
	SR [2]	✗	✓
	Z-Score [33]	✗	✗
	OC-SVM [34]	✓	✗
	MAD [35]	✗	✗
Multivariate	xStream [36]	✓	✗
	RShash [37]	✗	✗
	HSTree [38]	✗	✗
	LODA [39]	✓	✗
	RRCF [40]	✗	✗

However, comparing the existing anomaly detection benchmarks, we have found that it is challenging for them to meet all required properties. Therefore, we propose StreamAD, which aims to fill this gap and serve as the benchmark for anomaly detection on cloud platform metrics.

4. StreamAD: benchmark details

4.1. Overview

Streaming data refer to an infinite sequence of discrete data points that are continuously generated at a constant rate over time, donated as $\mathcal{X} = \{x_1, x_2, x_3, \dots, x_t, \dots\}$, where x_t represents the data point generated at time t. Compared to tabular data and static time series data, streaming data do not have a predetermined length. It refers to a continuous flow of data points arriving in real time. Based on this, online anomaly detection for streaming data can be defined as the process of identifying data points or patterns in a data streams that significantly deviate from the expected behaviors.

StreamAD is proposed for metrics online anomaly detection and its framework is shown in Fig. 3. It can ingest data streams directly from message queues like Kafka or RabbitMQ. Additionally, it provides a data stream generator that can simulate a streaming data environment using the loaded dataset. Each observation in the data stream can be formulated as (t, x_t), which represents an observation x_t with a timestamp t, note that x_t can be univariate or multivariate data. As each streaming data is observed, StreamAD first preprocesses the data. Typical data preprocessing methods include downsampling (aggregating data), scaling (transforming the data to a specific range), and normalization (scaling the individual samples to have unit norm).

The preprocessed data is then forwarded to the anomaly detection models. In StreamAD, there are eleven different detection algorithms as candidate models. The common methods of these algorithms are extracted as calculation operators, such as online statistics and sliding Fourier transformation [28–30]. These operators facilitate our analysis and detection of data streams in a continuous and online manner. The detection model assigns a score to each piece of data to reflect its anomaly degree. However, the output scores by different algorithms are on different scales due to their varying designs. Therefore, StreamAD provides score calibration methods that standardize the anomaly scores into a common scale and outputs them to the alert exporter. Since StreamAD is designed for online anomaly detection, when a data point is scored as normal by the detection model, the model should update itself based on the latest streaming data.

4.2. Anomaly detection algorithms

As detection models in StreamAD, anomaly detection algorithms play a crucial rule. They are the primary research objective in our benchmark. Numerous researchers [2,31,32,36–40] have contributed to the development of various algorithms, leading a prosperous research community. In pursuit of practical cloud platform metrics anomaly detection applications, StreamAD focuses on unsupervised online detection and has integrated eleven widely popular algorithms.

Although these algorithms rely on different techniques, they can be compared from two aspects. One aspect is the observation method of data streams. Some algorithms [31,34,36,39] observe the data stream through a sliding window. This kind of algorithms detect anomalies by comparing the differences between the latest window and the historical windows. While other methods [32,33,35,37,38,40] estimate the data distribution from historical data and examine whether the latest data point belongs to the distribution. The other aspect is the ability of different algorithms to capture the seasonal characteristics of a data stream. When a data pattern appears periodically in a data stream, it is usually regarded as normal. Some methods [2] can capture the seasonal patterns, while most distribution-based methods [32,33,35] cannot. In addition, these algorithms can be categorized into two types, namely univariate and multivariate, based on the data dimension that they can handle, as outlined in Table 2. The introduction of anomaly detection algorithms for both univariate and multivariate data streams are as follows.

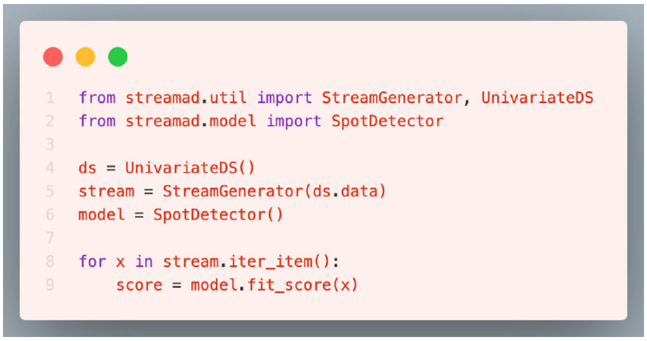
Univariate data streams anomaly detection refers to the algorithms that identify anomalies in the data streams containing only a single variable. These algorithms focus on analyzing the individual data. The advantages of these algorithms are their simplicity and nature intuition, as well as the great interpretability. KNN-CAD [31] is a sliding window-based method, it constructs a Hankel matrix to characterize the data streams and calculate the distance among observation windows using Mahalanobis distance. Streaming data that results in large distances may be potential anomalies. SPOT [32] and Z-Score [33] respectively assume normal streaming data conform to Pareto distribution and Gaussian distribution. Observations that fall out of the distribution are regarded as anomalies. SR [2] leverages Fourier transform to convert data streams from time domain to frequency domain, simplifying the anomaly detection task to identifying rare frequencies. OC-SVM [34] regards normal streaming data as belonging to the same class. It uses the support vector machine to determine the boundaries of normal data. Those data that cannot be classified as normal are considered anomalies. MAD [35] compares the deviation between newly arrived data and the median value of data stream histories. This detection algorithm has been proven to be effective in InfluxDB community [35].

Multivariate data streams anomaly detection identifies anomalies in data streams containing multiple variables. Different from univariate data streams, it takes into account the relationships, correlations and dependencies among various variables. This kind of anomaly detection algorithms provides us a comprehensive perspective, which enables the detection process to go beyond a specific metric and extend to a component within the cloud platform. For instance, we can simultaneously input the CPU, memory, and network metrics of a virtual machine into the algorithm to obtain an anomaly score.

xStream [36] tackles anomaly detection tasks for feature-evolving streams. As a density-based ensemble anomaly detection algorithm, it approximates the density of a point by counting its nearby neighbors at multiple scales. RShash [37] employs randomized hashing to score data points and features an elegant subspace interpretation. HSTree [38] is a fast one-class anomaly detector for evolving data streams. Utilizing mass [41] as a measure to rank anomalies, it can construct a ranking with small samples, enabling the anomaly detector to learn quickly and adapt to changes in data streams promptly. LODA [39] recognizes that the probability of observed samples valuable in determining their anomalousness. It approximates the joint probability using a collection of one-dimensional histograms, while each constructed on an input space projected onto a randomly generated vector. The use of one-dimensional histograms allows for efficient construction in one pass over the data, with simple query operations needed during classification. RRCF [40] introduces a robust random cut data structure that can serve as a sketch or synopsis of the input stream. This sketch can be efficiently updated in a dynamic data stream environment.

The above discussion presents a quick overview of eleven anomaly detection algorithms for both univariate and multivariate data streams. Some of these algorithms detect anomalies within a sliding observation window, while others incrementally update the detection model based on arriving data. Moreover, various algorithms have different capabilities in capturing seasonal patterns in data streams. Table 2 illustrates that the SR algorithm, which transforms data streams from the temporal domain into the frequency domain, can effectively detect anomalies in data streams that exhibit periodic features.

StreamAD offers a user-friendly API to access the aforementioned algorithms. Fig. 4 provides an usage example for SPOT anomaly detector, which is also the benchmark code snippet. The example code loads a benchmark dataset and simulates a stream environment using a loop. After that, the detector fits and scores each piece of data. The example illustrates how StreamAD assist users in evaluating their own use cases.



```python
from streamad.util import StreamGenerator, UnivariateDS
from streamad.model import SpotDetector

ds = UnivariateDS()
stream = StreamGenerator(ds.data)
model = SpotDetector()

for x in stream.iter_item():
    score = model.fit_score(x)
```

Fig. 4. API example for SPOT anomaly detector.

4.3. Incremental calculation operators

Due to the requirement for online detection of cloud platform metrics, an important property of online calculation is the incremental updating scheme of detection models. It differs significantly from offline calculation. In StreamAD, we extract the common online calculation methods of these algorithms as operators, which can help to enhance the efficiency of algorithm development.

A series of the operators are used for statistics. Take the variance calculation operator [42] as an example. A naive formula for calculating the variance $\sigma^{2,offline}$ of an offline dataset is:

$$\begin{aligned}\mu^{offline} &= \frac{1}{n}\sum_{i=1}^{n} x_i \\ \sigma^{2,offline} &= \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu^{offline})^2\end{aligned} \tag{1}$$

where n is the size of dataset, and $\mu^{offline}$ represents the mean value of the dataset. However, consider the online detection is under an infinite data stream setting, we cannot store all the history of data stream. Thus, we use the Welford's online algorithm [43] to handle the online calculation. For each incoming x, the variance incrementally updates as:

$$\begin{aligned}n &= n + 1 \\ \mu_{i+1}^{online} &= \mu_i + \frac{x - \mu_i^{online}}{n} \\ s_{i+1} &= s_i + (x - \mu_i) \times (x - \mu_{i+1}) \\ \sigma_{i+1}^{2,online} &= \frac{s_{i+1}}{n}\end{aligned} \tag{2}$$

where μ_{i+1}^{online} is the mean value of the first $i+1$ data from a stream, and s is the running sum of squares.

StreamAD has already included seven calculation operators for data statistics, and one operator for sliding Fourier transformation [28–30]. These incremental calculation operators play a vital role in efficiently processing data streams and updating models in real-time. By incorporating these calculation operators, StreamAD provides a comprehensive toolkit for developing and implementing new online anomaly detection algorithms, catering to the dynamic nature of data streams and the evolving requirements of online anomaly detection tasks.

4.4. Selection of datasets

The selection of datasets could have significantly impact on the benchmark results, as datasets from different domains exhibit varying

Table 3
Comparison of anomaly detection datasets across various properties in StreamAD.

Properties/Datasets	# Series	Avg. length	Anomaly ratio	Avg. span (# h)	Dimension	Metrics object
AIOPS_KPI [8]	29	103,588	2.65%	20.88	Univariate	container_cpu, queue, db, ping_time
MICRO [8]	29	228	1.26%	5.96	Univariate	oracle, container, docker, redis, linux
AWSCloudwatch [20]	17	3,984	0.05%	20.06	Univariate	cpu, network, request, grok, rds
GAIA [44]	279	10,156	0.78%	18.21	Univariate	zookeeper, redis, mysql
SMD [45]	28	25,300	4.16%	–	Multivariate	server machine instance

characteristics. Thus, we have chosen 5 public datasets that primarily focus on cloud platforms.

AIOPS_KPI [8] is a large-scale real-world public dataset, consisting of 27 key performance indicators (KPIs) for artificial intelligence for IT operations (AIOps). This dataset is collected from five large internet companies, including Sougo, eBay, Baidu, Tencent and Alibaba. The duration of each KPI data ranges from two to seven months, and each KPI is labeled by experienced SREs in these companies. The KPI patterns in this dataset are various.

MICRO [8] consists of metrics data from a public microservices monitoring dataset. It contains fine-grained metrics, including container, Linux system, Oracle, and Redis. The attributes of the spans on each component are aggregated into KPIs that reflect the overall status of each component. Anomalies in this dataset are simulated by fault injection (e.g., database close, container CPU stress, etc.), with labeled data recorded as fault injection timestamps.

AWSCloudwatch [20] features AWS server metrics collected by the Amazon Cloudwatch service. Example metrics include CPU Utilization, Network Bytes In, and Disk Read Bytes. This is a real-world dataset which shows us the behavior of AWS server.

GAIA [44] is comprised of one-month cloud platform monitoring data, selected from a login-action scenario in a business cloud platform system. The monitoring data includes Zookeeper, Redis and MySQL. This dataset covers different types of time series data, including change point, concept drift, periodic and stationary data. This dataset with rich variety of anomaly types can provide more comprehensively validation scenario for anomaly detection.

SMD [45] is a five-week dataset from a large Internet company, encompassing 28 different server machines. The data for each machine is divided into two equal-length segments for training and testing. It also provides labels indicating whether a point is an anomaly and the dimensions that contribute to each anomaly.

Table 3 presents a comparison of five anomaly detection datasets used in StreamAD, considering their various properties such as the number of instances, average length, anomaly ratio, average span, and metrics objects. The diversity of these selected datasets enables StreamAD to comprehensively evaluate anomaly detection algorithms.

4.5. Evaluation criteria

For time series anomaly detection evaluation, there are several measures have been proposed to assess the quality of anomaly detection algorithms. In general, these evaluation criteria can be classified into two categories, point-aware evaluation and series-aware evaluation. For these evaluation methods, precision and recall are both considered as evaluation criteria. Precision measures the proportion of relevant instances among the retrieved instances, and recall measures the proportion of relevant instances that were successfully retrieved. StreamAD includes both point-aware and series-aware evaluations to ensure a comprehensive assessment of the detection methods.

4.5.1. Point-aware evaluation

The point-aware evaluation criteria treats time series data as a collection of static data points, considering each point individually, as Fig. 5 shows. To perform the point-wise evaluation, let P and N represent the number of actual positive and negative points, while TP, FP, TN, and FN denote true positive, false positive, true negative, and false negative classifications, respectively. The following metrics are then defined:

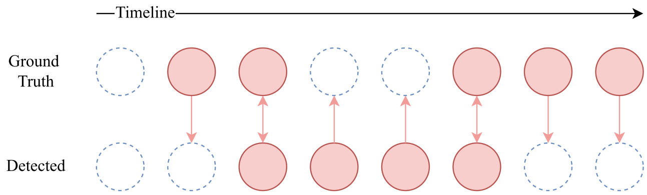


Fig. 5. Example of point-aware evaluation.

$$
\begin{aligned}
Precision^P &= \frac{TP}{TP+FP} \\
Recall^P &= \frac{TP}{P} = \frac{TP}{TP+FN}
\end{aligned}
\tag{3}
$$

Based on Eq. (3), the point-aware balanced F_1^P score is calculated as the harmonic mean of the $Precision^P$ and $Recall^P$, as:

$$
F_1^P = 2 \times \frac{Precision^P \times Recall^P}{Precision^P + Recall^P} \tag{4}
$$

Point-aware evaluation criteria are commonly employed in the literature for assessing the performance of anomaly detection algorithms. This approach offers a simple way to compare different methods in terms of their ability to identify individual anomalous data points within a time series.

However, point-aware evaluation exhibits certain limitations. By treating time series data as an uncorrelated set of individual data points, it overlooks the inherent temporal dependencies and relationships within the data, leading to a less accurate understanding of an algorithm's performance in capturing the underlying patterns and dynamics of the time series. Furthermore, point-aware evaluation overemphasizes the ability of algorithms to identify overall labeled anomalies, instead of considering more practical, application-specific concerns like cloud platform metric alerts. In real-world applications, SREs tend to focus on accurately detecting the starting positions of abnormal fragments, which are crucial for timely alerting and effective incident response. Point-aware evaluation may not adequately address this aspect, necessitating alternative evaluation methods that consider the practical requirements and goals of cloud platform monitoring and anomaly detection.

4.5.2. Series-aware evaluation

To alleviate shortcomings of the traditional Precision and Recall measures for time series anomaly detection, researchers [20,46–48] have proposed extensions for series-aware evaluation. The key insight behind series-aware evaluation is that an anomalous segment usually represents a single anomaly event, which may encompass multiple labeled anomaly points. In this context, anomalies at different positions within the segment owing varying weights. By considering anomaly events as continuous segments rather than isolated points, series-aware evaluation provides a more holistic assessment of an algorithm's ability to detect anomalies, accounting for temporal dependencies, and practical alerting considerations in real-world cloud platform applications.

Fig. 6 shows a typical example of series-aware evaluation. In this case, the detected anomalies may partially overlap with the ground

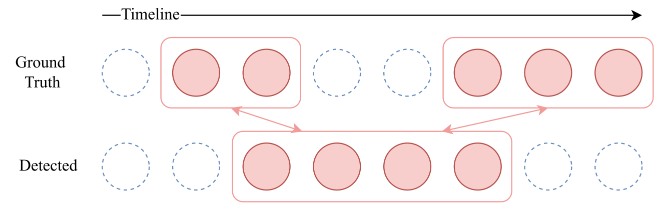


Fig. 6. Example of series-aware evaluation.

truth. Despite the small number of overlapping points lead to a low point-aware evaluation score, the detection still successfully identified the two anomalous sequence fragments, which holds practical value for cloud platform metrics monitoring. Additionally, considering the timeliness of alerts for metric anomalies, algorithms that detect the earlier portion of an anomaly sequence are preferred. Such early detection can help reduce the time required to respond to and address these anomalies. For example, in Fig. 6, the detection results that identify the true anomalies for the second ground truth sequence carry greater practical significance than those for the first sequence due to their earlier identification of the problematic segment. This aspect demonstrates the advantage of series-aware evaluation in assessing anomaly detection algorithms in practical applications.

Based on the above observations, we follow the idea from [46] and set series-aware evaluation criteria in StreamAD. Given a set of ground truth anomaly segments $R = \{R_1, .., R_{N_r}\}$ and a set of detected anomaly segments $P = \{P_1, .., P_{N_p}\}$, the $Precision^S$ and $Recall^S$ are defined as

$$\begin{aligned} Precision^S &= \frac{1}{N_P} \sum_{i=1}^{N_p} C(P_i, R) \times \sum_{j=1}^{N_r} \omega(P_i, P_i \cap R_j, \sigma) \\ Recall^S &= \frac{1}{N_r} \sum_{i=1}^{N_r} [\alpha E(R_i, P) + \\ &\quad (1-\alpha) C(R_i, P) \times \sum_{j=1}^{N_p} \omega(R_i, R_i \cap P_j, \sigma)] \end{aligned} \tag{5}$$

where $C(\cdot)$ is the cardinality factor, which is used for scaling the rewards earned based on the overlap size and position of detected anomalies. $E(\cdot)$ represents the existing reward function, which encourages to detect every anomaly segments. In addition, α, ω, σ serve as hyperparameters that depend on the specific practical applications. For the evaluation of cloud platform metric monitoring scenarios, these parameters are selected to prioritize early detection, accommodating that the front-end bias is often observed in such contexts.

Based on Eq. (5), the series-aware balanced F_1^S score is calculated as the harmonic mean of the $Precision^S$ and $Recall^S$, as:

$$F_1^S = 2 \times \frac{Precision^S \times Recall^S}{Precision^S + Recall^S} \tag{6}$$

Compared to point-aware evaluation, series-aware evaluation can yield more informative evaluation of anomaly detection algorithms in terms of their real-world performance and utility.

5. Experimental results

In this section, we provide a comprehensive analysis of our benchmark results, addressing various aspects of the anomaly detection algorithms. Firstly, we describe the experimental settings, encompassing the configurations of datasets, hyperparameters, evaluation criteria, and the evaluation platform. Next, we present in-depth results aiming at addressing the following questions:

1. How effective are the anomaly detection algorithms across different datasets?
2. Can the efficiency of the anomaly detection algorithms satisfy the requirements of practical applications?
3. Do the space complexities of detection algorithms remain static?

5.1. Experiment setting

Datasets. As described in Section 4.4, StreamAD includes five public real-world datasets, focusing specifically on cloud platform applications. Due to the online detection paradigm employed by various detection algorithms, we allocate the first one hundred points of each streaming data for algorithm initialization. These detection algorithms are primarily designed for the transductive setting, and outputting anomaly scores for the incoming data.

Hyperparameters. Each anomaly detection algorithm in StreamAD (described in Section 4.2) has its own hyperparameters, such as the observation window for KNN-CAD algorithm and the number of trees for RRCF algorithm. To ensure a fair comparison, we used the default hyperparameter settings from the original papers for all algorithms in StreamAD.

Evaluation criteria. The benchmark of StreamAD incorporates both point-aware and series-aware evaluation methods. The point-aware evaluation adheres to the standard definition of evaluation criteria, while the series-aware evaluation accounts for the front-end bias introduced in cloud platform metric evaluation scenarios, as discussed in [46].

Evaluation platform. All experiments are conducted on a server with the following configurations: Intel(R) Xeon(R) Platinum 8260 CPU @ 2.40 GHz, 16 cores, 32 GB RAM. The server runs Debian GNU/Linux 10 (64-bit). All the code is implemented with Python 3.8.

5.2. Benchmark effectiveness evaluation

The effectiveness of an algorithm plays a critical role in identifying anomalies accurately from data streams. As introduced in Section 4.5, we use point-aware and series-aware $Precision^{P/S}$, $Recall^{P/S}$ and $F_1^{P/S}$ as criteria to evaluate the effectiveness of various algorithms.

Table 4 shows the details of evaluation results. We conduct effectiveness experiments on all algorithms on univariate datasets, including AIOPS_KPI, MICRO, AWSCloudWatch, and GAIA. We also test the multivariate algorithms on the SMD dataset. The results indicate that no algorithm consistently exhibits high performance across all datasets, and the effectivenesses varies significantly. For instance, the Z-Score algorithm has the highest F_1^P score on the MICRO dataset but performs poorly on other datasets. Overall, xStream has a great $Recall$, it wins seven times out of ten experiments, which indicates that it can alert most true anomalies. On the other hand, the $Precision$ of SPOT ranks first, it wins six times out of ten experiments, indicating that most of the alerts generated by SPOT are true anomalies.

Additionally, point-aware evaluation and series-aware evaluation lead to significantly different results on the MICRO dataset. This is attributed to the short and concentrated anomaly duration of the MICRO dataset, as its average length is 228 and the anomaly rate is 1.26%. The experiments on this dataset demonstrate the differences between point-aware and series-aware evaluations.

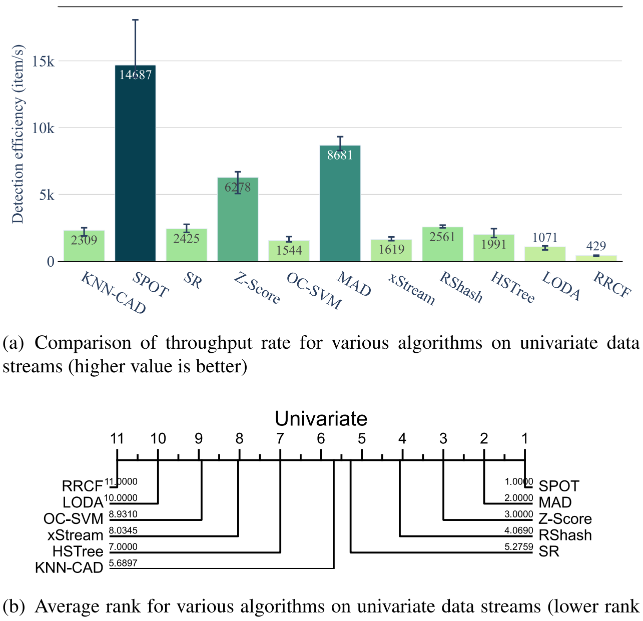
According to the experimental results, it is noteworthy that there is still a considerable scope for effectiveness improvement in these algorithms. The limitations of logical designs of the algorithms impact their performance, and some known flaws [7] in the existing datasets, such as unrealistic anomaly density and mislabeled ground truth, also have an unignorable impact.

In summary, selecting the best algorithm to detect cloud platform metrics anomalies is a challenging task. It is difficult to guarantee that an algorithm can cover all application scenarios. Users still need to try it out based on their specific use cases. As the promising effectiveness from our benchmark results, we recommend using SPOT for univariate data streams and xStream for multivariate data streams as the first attempt method.

Table 4
Effectiveness comparison of anomaly detection algorithms across various datasets in StreamAD.

Datasets	AIOPS_KPI			MICRO			AWSCloudWatch			GAIA			SMD		
							Point-aware evaluation								
Criteria/Algorithms	P^P	R^P	F_1^P	P^P	R^P	F_1^P	P^P	R^P	F_1^P	P^P	R^P	F_1^P	P^P	R^P	F_1^P
KNN-CAD	0.24	0.24	0.19	0.52	0.62	0.54	0.04	0.66	0.06	0.21	0.40	0.18		–	
SPOT	**0.44**	0.06	0.08	0.32	0.39	0.33	**0.07**	0.70	**0.12**	**0.37**	0.47	**0.29**			
SR	0.10	0.17	0.11	0.24	0.36	0.27	0.02	0.76	0.04	0.10	0.56	0.11			
Z-Score	0.23	0.15	0.13	**0.62**	0.56	**0.58**	0.04	**0.79**	0.07	0.10	0.63	0.11			
OC-SVM	0.13	0.26	0.14	0.50	0.63	0.54	0.02	0.77	0.03	0.13	0.68	0.15			
MAD	0.32	0.24	**0.22**	0.58	0.54	0.56	0.04	0.72	0.06	0.23	0.67	0.23			
xStream	0.05	**0.27**	0.06	0.11	**0.79**	0.17	0.01	**0.79**	0.01	0.01	**0.75**	0.01	0.04	0.21	0.06
RShash	0.18	0.18	0.15	0.06	0.39	0.08	0.02	0.75	0.04	0.14	0.63	0.17	0.06	0.02	0.02
HSTree	0.14	0.16	0.11	0.04	0.17	0.06	0.01	0.50	0.01	0.12	0.72	0.08	**0.19**	**0.10**	**0.11**
LODA	0.07	0.19	0.08	0.45	0.50	0.47	0.02	0.40	0.03	0.09	0.44	0.09	0.09	0.04	0.05
RRCF	0.14	0.13	0.11	0.56	0.39	0.33	0.04	0.73	0.06	0.12	0.55	0.12	0.13	0.05	0.06
							Series-aware evaluation								
Criteria/Algorithms	P^S	R^S	F_1^S	P^S	R^S	F_1^S	P^S	R^S	F_1^S	P^S	R^S	F_1^S	P^S	R^S	F_1^S
KNN-CAD	0.17	**0.33**	0.18	0.88	0.89	0.87	0.02	0.66	0.04	0.21	0.44	0.18		–	
SPOT	**0.43**	0.09	0.13	0.84	0.86	0.85	**0.06**	0.70	**0.11**	**0.37**	0.49	**0.28**			
SR	0.08	0.31	0.12	0.82	0.84	0.82	0.02	0.76	0.03	0.08	0.62	0.09			
Z-Score	0.18	0.23	0.15	0.89	0.89	0.89	0.04	**0.79**	0.06	0.08	0.73	0.09			
OC-SVM	0.07	**0.33**	0.09	0.86	0.91	0.87	0.01	0.77	0.02	0.10	0.76	0.12			
MAD	0.25	0.25	**0.21**	**0.93**	**0.92**	**0.93**	0.03	0.72	0.05	0.18	0.70	0.18			
xStream	0.03	0.23	0.05	0.67	**0.92**	0.70	0.01	**0.79**	0.01	0.04	0.74	0.02	0.04	**0.14**	0.05
RShash	0.15	0.22	0.15	0.70	0.81	0.71	0.02	0.75	0.04	0.14	0.67	0.17	0.05	0.01	0.01
HSTree	0.06	0.07	0.01	0.89	0.90	0.89	0.01	0.50	0.01	0.10	**0.77**	0.07	**0.15**	0.05	0.06
LODA	0.06	0.10	0.06	0.89	0.91	0.90	0.02	0.40	0.03	0.08	0.41	0.06	0.08	0.05	0.06
RRCF	0.10	0.27	0.12	0.88	0.85	0.82	0.03	0.73	0.04	0.10	0.61	0.11	0.10	0.12	**0.09**

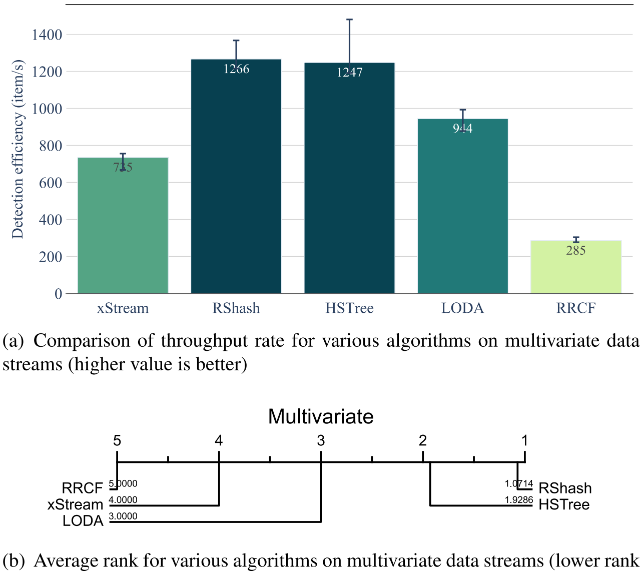
* *P* and *R* are abbreviations for *Precision* and *Recall*, respectively.
** Not conduct univariate algorithms experiments for multivariate data streams.



(a) Comparison of throughput rate for various algorithms on univariate data streams (higher value is better)

(b) Average rank for various algorithms on univariate data streams (lower rank is better)

Fig. 7. Efficiency comparison for eleven algorithms on univariate data streams.



(a) Comparison of throughput rate for various algorithms on multivariate data streams (higher value is better)

(b) Average rank for various algorithms on multivariate data streams (lower rank is better)

Fig. 8. Efficiency comparison for five algorithms on multivariate data streams.

5.3. Benchmark efficiency evaluation

For online anomaly detection in cloud platform, there are numerous metrics to be monitored. It is crucial to consider the efficiency of the detection methods in terms of the time required to respond to anomalous events, i.e., the execution time. The efficiency comparison provides a valuable assessment for the performance of various detection algorithms, which can help users to find.

Although different experimental environments, especially different computational resources, can greatly affect the implementation efficiency of algorithms, we believe that the horizontal comparison of different algorithms on the same experimental platform still has great significance. It allows us to intuitively compare the efficiency advantages and disadvantages of different algorithms.

In order to evaluate the efficiency of various algorithms across a range of existing datasets, we evaluate the number of detected streaming data per second for each algorithm, which provides us with an insight into the throughput rate. A higher throughput rate is generally

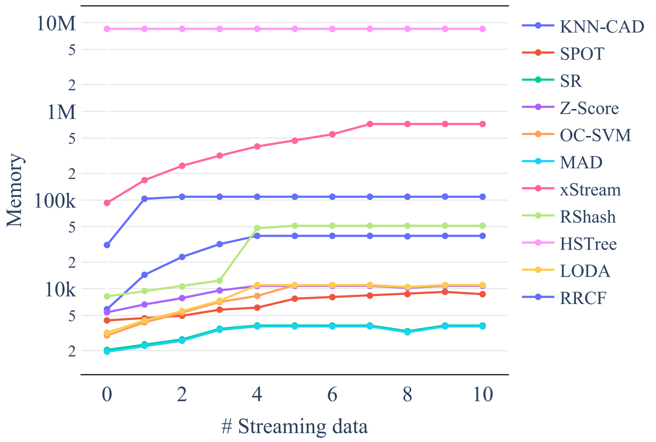


Fig. 9. Memory resource usage during online detection.

indicative of a more efficient algorithm, which is capable of addressing the anomaly detection task in large-scale cloud platforms.

Fig. 7 presents the efficiency comparison results of eleven algorithms for univariate data streams. In Fig. 7(a), we list the average throughout rate with the upper and lower bounds for various algorithms. It can be observed that SPOT outperforms other algorithms in this efficiency experiment, processing 14,687 data points per second. This exceptional performance can be attributed to its underlying assumption that most of the data is normal and does not require model fitting. Conversely, RRCF demonstrates the slowest performance among these algorithms due to its requirement to adjust each basic unit, i.e., random cut tree, when new streaming data arrives. It is a time-consuming process that slows down the overall detection performance. In this efficiency experiment, the fastest algorithm displays a significant performance advantage, processing data over 30 times faster than the slowest algorithm. The efficiency of the other algorithms varies, and their rankings based on performance are summarized in Fig. 7(b).

Additionally, we conducted the efficiency evaluation on multivariate datasets, as illustrated in Fig. 8. Among these algorithms, RShash and HSTree demonstrate similar efficiency, which can process more than one thousand points per second. Compared to Fig. 7(a), we observed that even when employing the same algorithm, processing multivariate data streams is slower than detection in univariate data streams.

Upon analyzing the results of all efficiency experiments, it can be observed that even the worst-performing algorithm, RRCF, is capable of detecting hundreds of points per second. Although a higher throughput rate generally signifies better performance, it is essential to consider the context of the practical application. For cloud platform anomaly detection, each metric requires a corresponding detection model instance, and the typical collection interval is 30 s per point. In this context, all algorithms in StreamAD, including the least efficient ones, can effectively satisfy the requirements regarding detection efficiency, ensuring timely anomaly detection and response in real-world cloud platform monitoring scenarios.

5.4. Memory limitation evaluation

As cloud platform metrics should be detected in real-time, the anomaly detection algorithm needs to be deployed and run for a long time. It is essential to ensure that the memory resources required by the algorithm do not increase continuously with data streaming, i.e., the memory resources should have a static limitation.

To evaluate the memory usage of various algorithms, we record the memory usage for the first one hundred data points under a univariate data stream. The results in Fig. 9 demonstrate that all the algorithms in StreamAD do not enlarge the occupied space after their initialization. Among all these algorithms, the tree-based method HSTree consumes the highest amount of memory. This can be attributed to the initialization of a tree by the algorithm, which arranges the historical stream data within it, and the size of the tree impacts the memory consumption of HSTree. We also find that the MAD algorithm has the lowest memory requirement. This is because MAD only needs to keep statistical information of the historical data streams without retaining all the records, which makes it more lightweight than others.

Thus, we believe that they can all comply with the memory limitation requirements for online applications. These results encourage the practical application of anomaly detection algorithms on cloud platforms, without a worry about the memory consumption.

6. Future work

With the development of the online anomaly detection community, the construction of benchmarks is a long-term process. In our future work, we are going to follow the state-of-the-art work, and integrate them into StreamAD. In addition, we plan to evaluate benchmarks from more perspectives, such as the impact of hyperparameters on the detection performance of different algorithms, and the interpretability of various algorithms. We hope that these future work can provide a more comprehensive view for benchmark evaluation.

7. Conclusion

In this work, we propose StreamAD, a cloud metrics-oriented benchmark for unsupervised online anomaly detection. StreamAD comprises eleven anomaly detection algorithms and conducts comprehensive experiments on five existing public datasets. The benchmark includes comparisons for the effectiveness, efficiency and memory resource consumption for various algorithms. StreamAD is open-source, and it provides a user-friendly API to help SREs evaluate anomaly detection applications in their specific use cases. Researchers can even develop new algorithms with StreamAD, which can facilitate further research in this area.

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 in part by National Key R&D Program of China: 2020YFB1805603, and CAS-Austria Joint Project (171111KYSB20200001).

References

[1] E. Bisong, An overview of Google cloud platform services, in: Building Machine Learning and Deep Learning Models on Google Cloud Platform: A Comprehensive Guide for Beginners, A Press, 2019, pp. 7–10.

[2] H. Ren, B. Xu, Y. Wang, C. Yi, C. Huang, X. Kou, T. Xing, M. Yang, J. Tong, Q. Zhang, Time-series anomaly detection service at microsoft, in: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Association for Computing Machinery, 2019, pp. 3009–3017.

[3] D.T. Shipmon, J.M. Gurevitch, P.M. Piselli, S.T. Edwards, Time series anomaly detection; Detection of anomalous drops with limited features and sparse examples in noisy highly periodic data, 2017, arXiv arXiv:1708.03665.

[4] D. Sun, M. Fu, L. Zhu, G. Li, Q. Lu, Non-intrusive anomaly detection with streaming performance metrics and logs for DevOps in public clouds: A case study in AWS, IEEE Trans. Emerg. Top. Comput. 4 (2016) 278–289.

[5] J. Gao, X. Song, Q. Wen, P. Wang, L. Sun, H. Xu, RobustTAD: Robust time series anomaly detection via decomposition and convolutional neural networks, 2021, arXiv, arXiv:2002.09545.

[6] Q. Cheng, D. Sahoo, A. Saha, W. Yang, C. Liu, G. Woo, M. Singh, S. Saverese, S.C.H. Hoi, AI for IT operations (AIOps) on cloud platforms: Reviews, opportunities and challenges, 2023, arXiv, arXiv:2304.04661.

[7] R. Wu, E.J. Keogh, Current time series anomaly detection benchmarks are flawed and are creating the illusion of progress, IEEE Trans. Knowl. Data Eng. 35 (2023) 2421–2429.

[8] Z. Li, N. Zhao, S. Zhang, Y. Sun, P. Chen, X. Wen, M. Ma, D. Pei, Constructing large-scale real-world benchmark datasets for AIOps, 2022, arXiv, arXiv:2208.03938.

[9] M. Ma, S. Zhang, D. Pei, X. Huang, H. Dai, Robust and rapid adaption for concept drift in software system anomaly detection, in: 2018 IEEE 29th International Symposium on Software Reliability Engineering, ISSRE, 2018, pp. 13–24.

[10] Z. Li, Y. Zhao, J. Han, Y. Su, R. Jiao, X. Wen, D. Pei, Multivariate time series anomaly detection and interpretation using hierarchical inter-metric and temporal embedding, in: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, Association for Computing Machinery, 2021, pp. 3220–3230.

[11] S. Han, X. Hu, H. Huang, M. Jiang, Y. Zhao, ADBench: Anomaly detection benchmark, Adv. Neural Inf. Process. Syst. 35 (2022) 32142–32159.

[12] 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 Endowment 15 (2022) 1697–1711.

[13] K.-H. Lai, D. Zha, G. Wang, J. Xu, Y. Zhao, D. Kumar, Y. Chen, P. Zumkhawaka, M. Wan, D. Martinez, X. Hu, TODS: An automated time series outlier detection system, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35, 2021, pp. 16060–16062.

[14] M. Pelc, Y. Khoma, V. Khoma, ECG signal as robust and reliable biometric marker: Datasets and algorithms comparison, Sensors 19 (2019) 2350.

[15] J. Wilkins, P. Seetharaman, A. Wahl, B. Pardo, Vocalset: a singing voice dataset, 2018.

[16] Google Cloud metrics, https://cloud.google.com/monitoring/api/metrics_gcp.

[17] B. Rabenstein, J. Volz, Prometheus: A Next-Generation Monitoring System (Talk), USENIX Association, 2015.

[18] Apache SkyWalking, https://skywalking.apache.org/.

[19] K.-H. Lai, D. Zha, J. Xu, Y. Zhao, G. Wang, X. Hu, Revisiting time series outlier detection: Definitions and benchmarks, in: Thirty-Fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1), 2022.

[20] A. Lavin, S. Ahmad, Evaluating real-time anomaly detection algorithms– the numenta anomaly benchmark, in: 2015 IEEE 14th International Conference on Machine Learning and Applications, ICMLA, 2015, pp. 38–44.

[21] N. Singh, C. Olinsky, Demystifying numenta anomaly benchmark, in: 2017 International Joint Conference on Neural Networks, IJCNN, 2017, pp. 1570–1577.

[22] V. Jacob, F. Song, A. Stiegler, B. Rad, Y. Diao, N. Tatbul, Exathlon: A benchmark for explainable anomaly detection over time series, Proc. VLDB Endowment 14 (2021) 2613–2626.

[23] D. Muhr, M. Affenzeller, Outlier/anomaly detection of univariate time series: A dataset collection and benchmark, in: Big Data Analytics and Knowledge Discovery, Springer International Publishing, 2022, pp. 163–169.

[24] S. Agrawal, J. Agrawal, Survey on anomaly detection using data mining techniques, Procedia Comput. Sci. 60 (2015) 708–713.

[25] V. Chandola, A. Banerjee, V. Kumar, Anomaly detection: A survey, ACM Comput. Surv. 41 (2009) 1–58.

[26] A. Boukerche, L. Zheng, O. Alfandi, Outlier detection: Methods, models, and classification, ACM Comput. Surv. 53 (2021) 1–37.

[27] M. Gupta, J. Gao, C.C. Aggarwal, J. Han, Outlier detection for temporal data: A survey, IEEE Trans. Knowl. Data Eng. 26 (2014) 2250–2267.

[28] E. Jacobsen, R. Lyons, The sliding DFT, IEEE Signal Process. Mag. 20 (2003) 74–80.

[29] E. Jacobsen, R. Lyons, An update to the sliding DFT, IEEE Signal Process. Mag. 21 (2004) 110–111.

[30] A. Chauhan, K.M. Singh, Recursive sliding DFT algorithms: A review, Digit. Signal Process. 127 (2022) 103560.

[31] E. Burnaev, V. Ishimtsev, Conformalized density- and distance-based anomaly detection in time-series data, 2016, arXiv, arXiv:1608.04585.

[32] A. Siffer, P.-A. Fouque, A. Termier, C. Largouet, Anomaly detection in streams with extreme value theory, in: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Association for Computing Machinery, 2017, pp. 1067–1075.

[33] Standard score, Wikipedia (2023).

[34] Y. Zhang, N. Meratnia, P. Havinga, Adaptive and online one-class support vector machine-based outlier detection techniques for wireless sensor networks, in: 2009 International Conference on Advanced Information Networking and Applications Workshops, 2009, pp. 990–995.

[35] A. Dotis-Georgiou, Anomaly detection with median absolute deviation, in: InfluxData.

[36] E. Manzoor, H. Lamba, L. Akoglu, Xstream: Outlier detection in feature-evolving data streams, in: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Association for Computing Machinery, 2018, pp. 1963–1972.

[37] S. Sathe, C.C. Aggarwal, Subspace outlier detection in linear time with randomized hashing, in: 2016 IEEE 16th International Conference on Data Mining, ICDM, 2016, pp. 459–468.

[38] S.C. Tan, K.M. Ting, T.F. Liu, Fast Anomaly Detection for Streaming Data.

[39] T. Pevný, Loda: Lightweight on-line detector of anomalies, Mach. Learn. 102 (2016) 275–304.

[40] S. Guha, N. Mishra, G. Roy, O. Schrijvers, Robust Random Cut Forest Based Anomaly Detection On Streams.

[41] K.M. Ting, G.-T. Zhou, F.T. Liu, J.S.C. Tan, Mass estimation and its applications, in: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Association for Computing Machinery, 2010, pp. 989–998.

[42] E. Schubert, M. Gertz, Numerically stable parallel computation of (Co-)variance, in: Proceedings of the 30th International Conference on Scientific and Statistical Database Management, ACM, 2018, pp. 1–12.

[43] B.P. Welford, Note on a method for calculating corrected sums of squares and products, Technometrics 4 (1962) 419–420.

[44] GAIA-DataSet/Companion_Data at Main · CloudWise-OpenSource/GAIA-DataSet, GitHub.

[45] Y. Su, Y. Zhao, C. Niu, R. Liu, W. Sun, D. Pei, Robust anomaly detection for multivariate time series through stochastic recurrent neural network, in: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, ACM, 2019, pp. 2828–2837.

[46] N. Tatbul, T.J. Lee, S. Zdonik, M. Alam, J. Gottschlich, Precision and recall for time series, in: Advances in Neural Information Processing Systems, Vol. 31, Curran Associates, Inc., 2018.

[47] W.-S. Hwang, J.-H. Yun, J. Kim, H.C. Kim, Time-series aware precision and recall for anomaly detection: Considering variety of detection result and addressing ambiguous labeling, in: Proceedings of the 28th ACM International Conference on Information and Knowledge Management, ACM, 2019, pp. 2241–2244.

[48] W.-S. Hwang, J.-H. Yun, J. Kim, B.G. Min, "Do you know existing accuracy metrics overrate time-series anomaly detections?", in: Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing, ACM, 2022, pp. 403–412.