ANALISIS KINERJA HADOOP DALAM PENGOLAHAN *BIG DATA* PADA *CLOUD COMPUTING* MENGGUNAKAN METODE BENCHMARK

SKRIPSI

Diajukan Untuk Melengkapi Salah Satu Syarat

Memperoleh Gelar Sarjana Komputer

OLEH :

ULLY AFIFA

09011282025093

JURUSAN SISTEM KOMPUTER

FAKULTAS ILMU KOMPUTER

UNIVERSITAS SRIWIJAYA

2025

HALAMAN PENGESAHAN

SKRIPSI

ANALISIS KINERJA HADOOP DALAM PENGOLAHAN BIG DATA PADA CLOUD COMPUTING MENGGUNAKAN METODE BENCHMARK

Sebagai salah satu syarat untuk penyelesaian studi di

Program Studi S1 Sistem Komputer

Oleh:

ULLY AFIFA

09011282025093

Pembimbing 1 : **Dr. Ahmad Heryanto, S. Kom, M.T**
NIP. 198701222015041002

Mengetahui

Ketua Jurusan Sistem Komputer

Dr. Ir. Sukemi, M.T
196612032006041001

HALAMAN PERSETUJUAN

Telah diuji dan lulus pada:

Hari : Jumat

Tanggal : 14 Februari 2025

Tim Penguji :

1. **Ketua : Aditya Putra Perdana Prasetyo, S.Kom., M.T**

2. **Penguji : Huda Ubaya, S.T., M.T.**

3. **Pembimbing : Dr. Ahmad Heryanto, S.Kom., M.T**

Mengetahui,

Ketua Jurusan Sistem Komputer

KEMENTERIAN PENDIDIKAN TINGGI, SAINS DAN TEKNOLOGI
UNIVERSITAS SRIWIJAYA
JURUSAN SISTEM KOMPUTER

Dr.Ir. Sukemi, M.T.

NIP.196612032006041001

HALAMAN PERNYATAAN

Yang bertanda tangan dibawah ini:

Nama : Ully Afifa
NIM : 09011282025093
Judul : Analisis Kinerja Hadoop Dalam Pengolahan *Big Data* Pada *Cloud Computing* Menggunakan Metode *Benchmark*

Hasil Pengecekan Software iThenticate/Turnitin: 6%

Menyatakan bahwa laporan tugas akhir saya merupakan hasil karya sendiri dan bukan hasil penjiplakan atau plagiat. Apabila ditemukan unsur penjiplakan atau plagiat dalam skripsi ini, maka saya bersedia menerima sanksi akademik dari Universitas Sriwijaya.

Demikian pernyataan ini saya buat dalam keadaan sadar dan tidak dipaksakan.

Indralaya, Maret 2025

Ully Afifa
NIM.09011282025093

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh

Segala puji dan syukur atas kehadirat Allah Subhanahu Wa Ta'ala yang telah memberikan karunia dan rahmat-Nya, sehingga penulis dapat menyelesaikan penulisan Tugas Akhir ini dengan judul **"**Analisis Kinerja Hadoop Dalam Pengolahan *Big Data* Pada *Cloud Computing* Menggunakan Metode *Benchmark*". Shalawat beriringkan salam senantiasa tercurahkan kepada Nabi Muhammad Sallallaahu 'Alaihi Wassalam yang telah membawa kedamaian dan rahmat untuk semesta alam serta menjadi suri tauladan bagi umatnya.

Dalam penulisan ini penulis menjelaskan mengenai bagaimana proses menganalisis kinerja Hadoop dalam pengolahan *Big Data* pada platform *Cloud Computing* menggunakan metode *benchmark*, di mana pengujian performa Hadoop dalam berbagai skenario pengolahan data besar menjadi tantangan menarik untuk dieksplorasi lebih lanjut.

Selesainya penulisan skripsi ini tidak terlepas dari peran serta semua pihak yang telah memberikan bantuan serta motivasi pada saat proses pembuatan laporan ini berlangsung. Oleh karena itu, pada kesempatan kali ini penulis menyampaikan ucapan terima kasih yang sebesar-besarnyua kepada:

1. Allah Subhanahu Wa Ta'ala, yang telah memberikan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan penulisan skripsi ini dengan baik dan lancar.
2. Kedua orang tua saya tercinta Bapak Usman Noviardi dan Ibu Sri haryati yang telah membesarkan, mendidik, mendukung saya serta tidak henti hentinya mendoakan dan memberikan nasihat, semangat, dan juga motivasi untuk saya dapat menghadapi segala sesuatu baik secara moril, materil, dan spiritual selama ini.

3. Ibu Rosmawati selaku keluarga yang telah memberikan tempat tinggal, kebutuhan primer, kebutuhan sekunder, dan kebutuhan tersier selama penulis menjalani masa perkuliahan hingga akhir.
4. Bapak Prof. Dr. Erwin, S.Si, M.Si., selaku Dekan Fakultas Ilmu Komputer Universitas Sriwijaya.
5. Bapak Dr. Ir. Sukemi, M.T., selaku Ketua Jurusan Sistem Komputer Fakultas Ilmu Komputer Universitas Sriwijaya.
6. Bapak Ahmad Heryanto, S. Kom, M.T., selaku Dosen Pembimbing Tugas Akhir yang telah berkenan meluangkan waktunya guna membimbing, memberikan saran dan motivasi serta bimbingan terbaik kepada penulis dalam menyelesaikan Tugas Akhir ini.
7. Bapak Muhammad Ali Buchari, M.T., selaku Dosen Pembimbing Akademik di Jurusan Sistem Komputer Fakultas Ilmu Komputer Universitas Sriwijaya.
8. Mba Sari selaku Admin Jurusan Sistem Komputer yang telah membantu dalam pengadministrasian.
9. Laboratorium *Comnets* sebagai tempat untuk berdiskusi dan menuangkan ide terkait permasalahan Tugas Akhir dan terima kasih atas infrastruktur lab yang digunakan dalam mengerjakan Tugas Akhir.
10. Sahara Diva Maharani dan Siti Triwinarti Ningrum yang telah bersedia menjadi teman dalam bertukar pikiran untuk menyelesaikan permasalahan pada Tugas Akhir ini.
11. Luqman Agus Dwiyono, M. Aziz Alhadi dan Ghulam Robbani Toha selaku teman-teman seperjuangan yang telah memberikan support dan bantuan selama penulis menjalani masa perkuliahan hingga akhir.
12. Dinia Tarisa Tuffahati dan Amanda selaku teman baik saya yang selalu menemani dan menghibur selama penyelesaian skripsi.
13. Teman-teman dari grup QUNA yang telah memberi dukungan kepada penulis selama penyelesaian skripsi.
14. Teman-teman kelas Sistem Komputer Unggulan angkatan 2020 yang sudah menghibur dan membantu penulis selama penyelesaian skripsi.
15. Seluruh pihak yang tidak dapat penulis sebutkan satu persatu, yang telah memberikan semangat serta do’a.

Penulis menyadari bahwa skripsi ini masih jauh dari kesempurnaan, oleh karena itu penulis sangat mengharapkan kritik dan saran yang bersifat membangun agar lebih baik lagi dikemudian hari.

Penulis berharap semoga skripsi ini dapat bermanfaat bagi pembaca, khususnya Fakultas Ilmu Komputer Universitas Sriwijaya secara langsung ataupun tidak langsung sebagai sumbang pikiran dalam peningkatan mutu pembelajaran dan dapat dijadikan referensi demi pengembangan ke arah yang lebih baik. Kebenaran datangnya dari Allah dan kesalahan datangnya dari diri penulis. Semoga Allah SWT senantiasa melimpahkan rahmat dan Ridho-Nya kepada kita semua.

Wassalamu'alaikum Warahmatullahi Wabarakatuh

Palembang, Maret 2025
Penulis,

Ully Afifa
NIM. 09011282025093

ANALISIS KINERJA HADOOP DALAM PENGOLAHAN *BIG DATA* PADA *CLOUD COMPUTING* MENGGUNAKAN METODE *BENCHMARK*

Ully Afifa (09011282025093)
Jurusan Sistem Komputer, Fakultas Ilmu Komputer, Universitas Sriwijaya
Email : u13lly@gmail.com

ABSTRAK

Pengolahan Big Data dalam Cloud Computing membutuhkan framework yang efisien, seperti Hadoop dan Spark. Penelitian ini menganalisis kinerja keduanya menggunakan metode benchmark dengan workload Terasort, menilai waktu eksekusi, throughput, serta penggunaan CPU dan memori. Pengujian dilakukan menggunakan workload Terasort pada infrastruktur cloud berbasis VirtualBox dengan mode cluster. Hasil penelitian menunjukkan bahwa Spark memiliki performa lebih unggul dibandingkan Hadoop, dengan waktu eksekusi yang lebih cepat hingga 4,7 kali lipat pada beban kerja tertentu dan peningkatan throughput sebesar 92,25% dibandingkan Hadoop. Namun, Spark mengonsumsi memori lebih tinggi, sekitar 10% lebih besar dibandingkan Hadoop. Di sisi lain, Hadoop menunjukkan efisiensi yang lebih baik dalam pemanfaatan sumber daya serta lebih stabil dalam kondisi beban kerja yang besar.Penelitian ini memberikan wawasan bagi pengguna dalam memilih platform pengolahan Big Data yang sesuai dengan kebutuhan spesifik mereka. Dengan memahami keunggulan dan keterbatasan masing-masing framework, implementasi Hadoop dan Spark dapat dioptimalkan untuk meningkatkan efisiensi dalam pemrosesan data berskala besar.

Kata kunci: *Big Data, Cloud Computing*, Hadoop, Spark, *Benchmark*, Terasort

HADOOP PERFORMANCE ANALYSIS IN BIG DATA PROCESSING ON CLOUD COMPUTING USING BENCHMARK METHOD

Ully Afifa (09011282025093)
Jurusan Sistem Komputer, Fakultas Ilmu Komputer, Universitas Sriwijaya
Email : u13lly@gmail.com

ABSTRACT

Big Data processing in Cloud Computing requires efficient frameworks such as Hadoop and Spark. This study analyzes their performance using the benchmark method with the Terasort workload, evaluating execution time, throughput, and CPU and memory usage. The testing was conducted using the Terasort workload on a cloud infrastructure based on VirtualBox in cluster mode. The results show that Spark outperforms Hadoop, with execution time up to 4.7 times faster for certain workloads and 92.25% higher throughput compared to Hadoop. However, Spark consumes 10% more memory than Hadoop. On the other hand, Hadoop demonstrates better resource efficiency and greater stability under heavy workloads. This study provides insights for users in selecting the appropriate Big Data processing platform based on specific needs. By understanding the advantages and limitations of each framework, the implementation of Hadoop and Spark can be optimized to enhance efficiency in large-scale data processing.

Key Words : *Big Data, Cloud Computing*, Hadoop, Spark, *Benchmark*, Terasort

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

BAB I
PENDAHULUAN

1.1. Latar Belakang

Perkembangan teknologi informasi yang pesat telah mendorong munculnya era data besar (big data), di mana volume data yang dihasilkan terus meningkat secara eksponensial. Data besar memiliki karakteristik khusus, seperti volume yang sangat besar, variasi yang beragam, kecepatan aliran data, nilai informasi, valensi hubungan data, serta kebenaran data (veracity). Karakteristik ini menciptakan tantangan tersendiri dalam proses pengelolaan, analisis, dan ekstraksi pengetahuan dari data tersebut. Salah satu tantangan utama adalah banyaknya data yang tidak terstruktur, seperti teks, gambar, audio, dan video, yang membutuhkan metode khusus untuk diproses menjadi informasi yang bermanfaat. Di sisi lain, keterbatasan perangkat keras dan perangkat lunak tradisional dalam menangani data berskala besar memerlukan inovasi baru, baik dari segi infrastruktur teknologi maupun metode analisis[1]. Meningkatnya *volume data* yang tersedia dalam beragam format, menyebabkan biaya penyimpanan data juga terus meningkat, sehingga akses dan penggunaan data menjadi tak terjangkau [2] [3].

Big Data menghadirkan peluang dan tantangan bagi berbagai sektor. Di satu sisi, *Big Data* membuka peluang untuk meningkatkan efisiensi, membuat keputusan strategis, dan mengembangkan solusi inovatif, tak terkecuali bagi sektor bisnis, sains, hingga pemerintahan. Namun, di sisi lain, *Big Data* juga menghadirkan tantangan dalam hal pengelolaan, analisis, dan keamanan data. Alat tradisional untuk mengolah data tak lagi cukup untuk menangani volume dan kompleksitas *Big Data* sehingga diperlukan pendekatan-pendekatan baru[4].

Cloud Computing telah menjadi teknologi utama dalam mendukung analisis big data yang skalabel, mengingat kebutuhan untuk menangani data dalam jumlah besar yang terus meningkat. Teknologi ini menyediakan *Platform as a Service* (PaaS), yang memungkinkan proses query dan analisis data besar menjadi lebih hemat biaya dan efisien. Dalam arsitektur TI terdistribusi ini, data pengguna diproses di tepi jaringan, sehingga menghilangkan kebutuhan akan perangkat keras,

ruang fisik, dan perangkat lunak yang mahal. *Cloud Computing* menawarkan fleksibilitas sumber daya, percepatan inovasi, dan skala ekonomi melalui layanan yang diberikan melalui internet. Konsep ini telah merevolusi infrastruktur komputasi dengan memperluas model layanan berbasis komputasi, seperti yang disediakan oleh penyedia besar seperti Amazon, Google, dan Microsoft, yang menawarkan sistem big data dengan biaya terjangkau[5].

Pemrosesan data dalam skala besar yang menggunakan algoritma seperti terasort akan membutuhkan waktu yang lama jika dijalankan secara berurutan, terutama ketika volume data sangat besar. Oleh karena itu, diperlukan solusi komputasi terdistribusi untuk meningkatkan efisiensi serta kecepatan pemrosesan data. Pendekatan utama dalam menangani permasalahan big data ini adalah melalui penggunaan kerangka kerja pengolahan data besar (*big data frameworks*) seperti Apache Hadoop dan Apache Spark[6].

Apache Hadoop adalah platform yang mampu mendistribusikan data dalam jumlah besar secara efektif melalui model pemrograman khusus. *Apache* Hadoop digunakan untuk menyediakan layanan pemrosesan dan penyimpanan data besar yang dapat digunakan dalam berbagai aplikasi, seperti analisis keuangan, analisis log, pembelajaran mesin, dan penyimpanan data. Hadoop terdiri dari beberapa komponen utama, yaitu *Hadoop Distributed File System* (HDFS) untuk penyimpanan data, MapReduce untuk model pemrograman, dan *Yet Another Resource Negotiator* (YARN) untuk manajemen sumber daya[7]. Sebaliknya, Apache Spark adalah kerangka kerja komputasi terdistribusi yang menawarkan pemrosesan dalam memori dan menggunakan Resilient Distributed Datasets (RDDs) yang bersifat *in-memory* untuk memproses data[8].

Meskipun ada kesamaan, terdapat perbedaan signifikan antara Apache Hadoop dan Apache Spark dalam hal arsitektur, kecepatan pemrosesan, skalabilitas, dan kemudahan penggunaan. Perbedaan ini memunculkan perdebatan tentang alat mana yang lebih cocok untuk kasus penggunaan analitik Big Data tertentu.

Pasar memprediksi pertumbuhan tahunan untuk teknologi Hadoop sebesar 16,10% dari tahun 2019 hingga 2029, dengan pertumbuhan tercepat di kawasan Asia-Pasifik dan pasar terbesar di Amerika Utara. Pada tahun 2023, skala pasar

global Hadoop telah mencapai 64,485 miliar yuan, dan diperkirakan akan meningkat menjadi 152,193 miliar yuan pada tahun 2029. Banyak perusahaan ternama, seperti IBM, Adobe, dan A9.com, serta perusahaan-perusahaan besar di Tiongkok seperti Baidu, Alibaba, Tencent, dan Huawei, telah menerapkan teknologi Hadoop dalam strategi big data mereka. Hadoop memainkan peran penting di berbagai bidang utama, seperti penambangan data, analisis log, data warehouse, dan mesin rekomendasi. Ke depan, Hadoop akan terus menjadi teknologi kunci dalam pengolahan big data dan memberi nilai tambah di banyak industri[9].

Contoh penerapan Hadoop dapat dilihat pada penelitian[10]. Penelitian ini memperkenalkan sebuah metodologi untuk membangun cluster Hadoop virtual pada cloud pribadi, dengan tujuan menemukan konfigurasi terbaik yang efisien untuk menjalankan aplikasi MapReduce skala besar. Metodologi ini mempertimbangkan beberapa faktor terkait Hyper-V, Hadoop, HDFS, dan infrastruktur penyimpanan SAN redundan yang digunakan. Khususnya, aspek seperti multipathing pada SAN, replikasi HDFS, dan kesadaran rak (rack awareness) dalam HDFS menjadi fokus utama. Eksperimen ini menunjukkan bahwa skala cluster virtual secara signifikan meningkatkan kinerja aplikasi MapReduce (*WordCount*) untuk beban kerja sebesar 750 GB, dibandingkan kinerja aplikasi yang sama pada cluster fisik dengan jumlah server yang serupa. Namun, peningkatan kinerja ini memiliki batas, karena penambahan jumlah VM di atas ambang tertentu tidak lagi memberikan perbaikan kinerja. Hal ini disebabkan oleh pemakaian penuh CPU dan kejenuhan pada koneksi komunikasi saat menangani jumlah operasi I/O bersamaan yang besar. Hasil eksperimen menunjukkan peningkatan kinerja aplikasi MapReduce, dengan kecepatan hingga 3,54 kali lebih baik pada aplikasi *WordCount* dengan beban kerja 750 GB.

Pada penelitian [11] membahas dan membandingkan Hadoop dan Spark, menjelaskan perbedaan keduanya dalam fase Map dan Reduce. Selanjutnya, dilakukan perbandingan kinerja antara kedua framework ini yang dijalankan dalam mode pseudo-distributed menggunakan berbagai beban kerja dari HiBench. *Wordcount* workload dijalankan dengan berbagai ukuran data untuk memperoleh hasil yang lebih akurat. Selain itu, hasil penerapan Hadoop pada Amazon EC2 turut

disajikan guna menunjukkan dampak cloud nyata terhadap kinerja framework Big Data. Berdasarkan evaluasi hasil eksperimen pada berbagai benchmark, diperoleh kesimpulan bahwa Spark menunjukkan pemanfaatan CPU dan beban sistem yang lebih rendah dibanding Hadoop.

Selain itu, Spark secara konsisten memiliki performa lebih unggul daripada Hadoop MapReduce, terutama pada algoritma iteratif seperti Pagerank yang menghasilkan peningkatan 18 kali lipat serta 4,7 kali pada beban kerja lainnya. Dari aspek throughput, Spark menunjukkan peningkatan sebesar 92,25% pada workload *Wordcount* dan 28,57% pada beban kerja lainnya jika dibandingkan dengan Hadoop. Namun, Spark mengonsumsi lebih banyak memori, yaitu sekitar 10% lebih tinggi daripada Hadoop. Faktor-faktor seperti pemrosesan data dalam memori (in-memory processing) berbasis Distributed Resilient Datasets (RDD) memberikan keunggulan Spark dalam optimasi akses disk, penggunaan bandwidth memori, dan tingkat IPC (Instructions Per Cycle), sehingga performanya lebih baik. Namun, karena Spark memuat dan menyimpan data dalam memori, kinerjanya dapat menurun jika memori tidak mencukupi dan berpotensi lebih lambat dari Hadoop dalam kondisi tertentu. Oleh karena itu, pada lingkungan dengan keterbatasan memori atau kecepatan bukan prioritas utama, Hadoop bisa menjadi pilihan yang lebih baik. Pengaturan cluster yang tepat, seperti jumlah Mapper dan Reducer, juga sangat memengaruhi waktu eksekusi pekerjaan.

Pada penelitian [12] mengembangkan kerangka cloud berbasis Hadoop untuk mengurangi beban komputasi dalam optimasi simulasi model. Hasil dari penelitian meningkatkan kecepatan kalibrasi model *Soil and Water Assessment Tool* (SWAT) hingga 55-58 kali lipat dengan 100 inti, dan kerangka dapat menangani kegagalan *node* tanpa mempengaruhi simulasi. Penelitian ini menunjukkan di mana *node* komputasi ditingkatkan atau dikurangi secara dinamis menunjukkan bahwa kerangka kerja tersebut dapat secara otomatis menyeimbangkan kembali beban kerja di seluruh *node* yang tersisa.

Lalu pada penelitian [13] ini menyelidiki batasan kinerja kerangka kerja Hadoop yang bergantung pada manajemen sumber daya statis berbasis *container*, serta menganalisis peningkatan kinerjanya. Pengaturan jumlah concurrent

containers (cc) per node dan ukuran blok HDFS yang optimal sangat mempengaruhi kinerja tahap map pada berbagai program MapReduce. Untuk program TeraSort, kinerja optimal pada tahap map dicapai ketika terdapat satu container per disk. Dengan cc optimal sebesar 1, program TeraSort meningkatkan kinerja tahap map sebesar 52,33% pada ukuran blok HDFS 128MB dan 59,16% pada ukuran blok 256MB. Demikian pula, program Sort dengan cc optimal sebesar 1 meningkatkan kinerja tahap map sebesar 47,90% dan 64,24% pada ukuran blok HDFS 128MB dan 256MB, masing-masing.

Namun, untuk pekerjaan CPU-bound seperti *WordCount*, kinerja tahap map yang optimal dicapai ketika cc mencapai 4 containers per node, karena prosesnya lebih banyak menggunakan CPU, dan mesin uji memiliki 4 vCores. Dalam pekerjaan yang membutuhkan banyak CPU, menggunakan satu map task per CPU memberikan kinerja optimal, sedangkan untuk pekerjaan yang membutuhkan banyak disk atau I/O, menjalankan satu map task per disk akan memberikan waktu yang paling efisien pada tahap map. Program *WordCount* dengan cc optimal sebesar 4 mampu meningkatkan kinerja tahap map sebesar 4,08% pada ukuran blok HDFS 128MB dan 18,05% pada ukuran blok 256MB.

Pada penelitian [14] menganalisis pendekatan caching hibrida cerdas baru yang menggabungkan keunggulan dari algoritma penggantian cache H-SVM-LRU dan kebijakan penjadwalan CLQLMRS. Hasil eksperimen menunjukkan bahwa pendekatan baru ini memberikan peningkatan waktu eksekusi sebesar 31,2%, yang disebabkan oleh rasio hit cache yang lebih tinggi berkat algoritma H-SVM-LRU, serta meningkatnya kemungkinan tugas menggunakan data lokal melalui kebijakan penjadwalan CLQLMRS.

Alasan utama penelitian ini dilakukan adalah adanya ketidakpastian tentang kemampuan Hadoop, kurangnya pemahaman tentang faktor yang mempengaruhi kinerja, serta kebutuhan mendesak akan solusi untuk mengoptimalkan kinerja Hadoop[15]. Dengan kemampuan integrasi yang baik antara berbagai alat dalam ekosistem Hadoop, analisis kinerja Hadoop memudahkan kita dalam menerapkan dan membandingkan hasil benchmark dalam konteks sistem yang lebih luas,

sehingga menjadikannya pilihan yang lebih efisien untuk melakukan analisis kinerja.

Penelitian ini bertujuan untuk menilai performa Hadoop dan Spark di lingkungan cloud dengan menggunakan metode benchmark. Evaluasi dilakukan melalui beban kerja terasort, yang merepresentasikan tugas umum dalam pemrosesan data. Analisis mencakup kinerja, seperti waktu eksekusi dan throughput, serta pemanfaatan sumber daya, termasuk CPU dan memori. Hasil penelitian ini diharapkan dapat memberikan wawasan mengenai keunggulan dan keterbatasan Hadoop serta Spark dalam pengolahan data. Penelitian ini akan menjadi panduan bagi praktisi dan peneliti di bidang big data dalam memilih platform yang sesuai untuk kebutuhan pemrosesan data. Dengan demikian, penulis mengusulkan penelitian dengan judul " Analisis Kinerja Hadoop Dalam Pengolahan *Big Data* Pada *Cloud Computing* Menggunakan Metode *Benchmark*"

1.2. Rumusan Masalah

Berdasarkan latar belakang yang telah dikemukakan sebelumnya, maka didapatkan perumusan masalah yaitu :

1. Bagaimana kinerja Hadoop dan Spark dalam mengolah *Big Data* pada lingkungan *Cloud Computing* dalam hal waktu eksekusi dan throughput saat menjalankan beban kerja Terasort?
2. Bagaimana penggunaan performa sumber daya (CPU dan memori) oleh Hadoop dan Spark saat menjalankan Terasort dengan berbagai ukuran data?

1.3. Tujuan

Berdasarkan rumusan masalah yang telah dibuat, penelitian ini bertujuan untuk:

1. Menganalisis kinerja Hadoop dan Spark dalam mengolah *Big Data* pada lingkungan *Cloud Computing* dalam hal waktu eksekusi dan throughput saat menjalankan beban kerja Terasort.

2. Mengidentifikasi performa penggunaan sumber daya (CPU dan memori) oleh Hadoop dan Spark saat menjalankan beban kerja Terasort dengan berbagai ukuran data.

1.4. Manfaat Penelitian

Penelitian ini diharapkan dapat memberikan manfaat sebagai berikut:

1. Memberikan informasi tentang kemampuan Hadoop dalam mengolah *Big Data* pada platform *Cloud Computing*.
2. Membantu pengguna dalam memilih platform pengolahan *Big Data* yang tepat untuk kebutuhan.
3. Meningkatkan kinerja *Big Data* pada platform *Cloud Computing*.

1.5. Batasan Penelitian

Adapun batasan penelitian yang terdapat dalam penyusunan skripsi ini, yaitu:

1. Perfoma pada penelitian ini berdasarkan pada waktu eksekusi, throughput, dan penggunaan sumber daya.
2. Penelitian ini akan fokus pada perbandingan performa antara Hadoop dan Spark dalam mode cluster dengan input data berupa teks.
3. Pembuatan dan evaluasi kinerja Hadoop dan Spark akan dijalankan pada infrastruktur cloud berbasis VirtualBox.

1.6. Sistematika Penulisan

Untuk dapat mempermudah dan memperjelas proses penyusunan Skripsi ini, dibuat sistematika penulisan sebagai berikut:

BAB I. PENDAHULUAN

Bab ini berisikan penjelasan secara sistematis berupa topik penelitian yang berisikan latar belakang, tujuan, manfaat, perumusan masalah, Batasan masalah, serta sistematika penulisan.

BAB II. TINJAUAN PUSTAKA

Bab ini berisikan penjelasan dasar teori dari penelitian mengenai *Big Data, Cloud Computing,* dan *Benchmark*

BAB III. METODOLOGI PENELITIAN

Pada bab ketiga, membahas proses yang dilakukan dalam penelitian secara sistematis. Serta mengkaji tahapan perancangan sistem, dan penerapan dari metode penelitian.

BAB IV. HASIL DAN ANALISA

Bab ini menjelaskan hasil dari proses pengujian yang telah dilakukan, dan melakukan analisis data yang didapat dari hasil pengujian.

BAB V. KESIMPULAN DAN SARAN

Pada bab terakhir, berisikan kesimpulan dan saran dari hasil analisa berdasarkan penelitian yang telah dilakukan.

DAFTAR PUSTAKA

[1] M. K. M. Nasution, O. S. Sitompul, M. Elveny, and R. Syah, "Data science: A Review towards the Big Data Problems," *J. Phys. Conf. Ser.*, vol. 1898, no. 1, 2021.

[2] D. C. K. Gomathy, "Bigdata Analysis on Methods and Tools," *Interantional J. Sci. Res. Eng. Manag.*, vol. 06, no. 11, 2022.

[3] H. K. Gupta and R. Parveen, "Comparative Study of Big Data Frameworks," *IEEE Int. Conf. Issues Challenges Intell. Comput. Tech. ICICT 2019*, no. February, 2019.

[4] A. K. Ni Komang, I. M. A. D. Suarjaya, and I. M. S. Raharja, "Classification of Public Figures Sentiment on Twitter using Big Data Technology," *J. Informatics Telecommun. Eng.*, vol. 6, no. 1, pp. 157–169, 2022.

[5] M. Das and R. Dash, "Role of cloud computing for big data: A review," *Smart Innov. Syst. Technol.*, vol. 153, no. September, pp. 171–179, 2021.

[6] N. Ahmed, A. L. C. Barczak, T. Susnjak, and M. A. Rashid, "A comprehensive performance analysis of Apache Hadoop and Apache Spark for large scale data sets using HiBench," *J. Big Data*, vol. 7, no. 1, 2020.

[7] K. J. Merceedi and N. A. Sabry, "A Comprehensive Survey for Hadoop Distributed File System," *Asian J. Res. Comput. Sci.*, no. August, pp. 46–57, 2021.

[8] H. Ahmadvand, M. Goudarzi, and F. Foroutan, "Gapprox: using Gallup approach for approximation in Big Data processing," *J. Big Data*, vol. 6, no. 1, 2019.

[9] P. Jiang, "Application Status of Hadoop in Data Cloud Computing," pp. 1–4, 2024.

[10] F. Al-Hawari, K. Tayem, S. Alouneh, and A. Al Ksasbeh, "Impact of Virtual Hadoop Cluster Scalability on The Performance of Big Data Mapreduce Applications," *2023 24th Int. Arab Conf. Inf. Technol. ACIT 2023*, no. December, 2023.

[11] Y. Samadi, M. Zbakh, and C. Tadonki, "Performance comparison between hadoop and spark frameworks using HiBench benchmarks," *Concurr.*

Comput. Pract. Exp., vol. 30, no. 12, 2018.

[12] J. Ma, K. Rao, R. Li, Y. Yang, W. Li, and H. Zheng, "Improved Hadoop-based cloud for complex model simulation optimization: Calibration of SWAT as an example," *Environ. Model. Softw.*, vol. 149, p. 105330, 2022.

[13] T. T. Htay and S. Phyu, "Improving the performance of Hadoop MapReduce Applications via Optimization of concurrent containers per Node," *2020 IEEE Conf. Comput. Appl. ICCA 2020*, pp. 1–5, 2020.

[14] R. Ghazali, D. G. Down, and G. Ca, "Overview of Caching Mechanisms to Improve Hadoop Performance," no. October, 2023.

[15] J. Zhang and M. Lin, "A comprehensive bibliometric analysis of Apache Hadoop from 2008 to 2020," *Int. J. Intell. Comput. Cybern.*, vol. 16, no. 1, pp. 99–120, Jan. 2023.

[16] M. Al-Hami, M. Maabreh, S. Taamneh, A. Pradeep, and H. B. Salameh, "Apache hadoop performance evaluation with resources monitoring tools, and parameters optimization: Iot emerging demand," *J. Theor. Appl. Inf. Technol.*, vol. 99, no. 11, pp. 2734–2750, 2021.

[17] K. L. Bawankule, R. K. Dewang, and A. K. Singh, "Performance analysis of hadoop YARN job schedulers in a multi-tenant environment on HiBench benchmark suite," *Int. J. Distrib. Syst. Technol.*, vol. 12, no. 3, 2021.

[18] F. Ullah, S. Dhingra, X. Xia, and M. A. Babar, "Evaluation of distributed data processing frameworks in hybrid clouds," *J. Netw. Comput. Appl.*, vol. 224, no. February, p. 103837, 2024.

[19] Y. Li, "Performance Analysis of Scheduling Algorithms in Apache Hadoop," *Proc. - 2020 16th Int. Conf. Comput. Intell. Secur. CIS 2020*, pp. 149–154, 2020.

[20] N. Ahmed, A. L. C. Barczak, S. U. Bazai, T. Susnjak, and M. A. Rashid, "Performance Analysis of Multi-Node Hadoop Cluster Based on Large Data Sets," *2020 IEEE Asia-Pacific Conf. Comput. Sci. Data Eng. CSDE 2020*, no. 2, 2020.

[21] M. H. P. Swari, I. K. S. Satwika, and I. P. S. Handika, "Performance analysis of sales big data processing using hadoop and hive in cloud environment," *Proceeding - 6th Inf. Technol. Int. Semin. ITIS 2020*, pp. 162–166, 2020.

[22] P. Auradkar, T. Prashanth, S. Aralihalli, S. P. Kumar, and D. Sitaram, "Performance tuning analysis of spatial operations on Spatial Hadoop cluster with SSD," *Procedia Comput. Sci.*, vol. 167, no. 2019, pp. 2253–2266, 2020.

[23] S. R. M. Zeebaree *et al.*, "Characteristics and Analysis of Hadoop Distributed Systems," *Int. J. Networked Distrib. Comput.*, vol. 4, no. 3, pp. 96112–96127, 2021.

[24] Y. Benlachmi, A. El Yazidi, and M. L. Hasnaoui, "A Comparative Analysis of Hadoop and Spark Frameworks using Word Count Algorithm," *Int. J. Adv. Comput. Sci. Appl.*, vol. 12, no. 4, pp. 778–788, 2021.

[25] S. Lim and D. Park, "Improving Hadoop MapReduce performance on heterogeneous single board computer clusters," *Futur. Gener. Comput. Syst.*, vol. 160, no. June, pp. 752–766, 2024.

[26] K. Kalia *et al.*, "Improving MapReduce heterogeneous performance using KNN fair share scheduling," *Rob. Auton. Syst.*, vol. 157, p. 104228, 2022.

[27] S. Yang, W. Jin, Y. Yu, and K. F. Hashim, "Optimized hadoop map reduce system for strong analytics of cloud big product data on amazon web service," *Inf. Process. Manag.*, vol. 60, no. 3, p. 103271, 2023.

[28] K. B. Naidu *et al.*, "Analysis of Hadoop log file in an environment for dynamic detection of threats using machine learning," *Meas. Sensors*, vol. 24, no. July, p. 100545, 2022.

[29] F. Al-Hawari, K. Tayem, S. Alouneh, and A. Al-Ksasbeh, "Methodology to Evaluate the Performance of Hadoop MapReduce on a Hyper-V Cluster using SAN Storage," *Proc. - 2022 23rd Int. Arab Conf. Inf. Technol. ACIT 2022*, no. November, 2022.

[30] P. Kalyanaraman, K. R. Jothi, P. Balakrishnan, R. G. Navya, A. Shah, and V. Pandey, "Implementing Hadoop Container Migrations in OpenNebula Private Cloud Environment," *Role Edge Anal. Sustain. Smart City Dev.*, pp. 85–103, 2020.

[31] T. T. Htay and S. Phyu, "Towards Performance Optimization for Hadoop MapReduce Applications," *17th Int. Conf. Electr. Eng. Comput. Telecommun. Inf. Technol. ECTI-CON 2020*, pp. 698–701, 2020.

[32] L. Awad, E. Tawfiq Naffar, L. Sadi Awad, F. Ahmad Alzaghoul, and K.

Abdullah, “Apache Spark and Hadoop: a Detailed Comparison of the Two Processing Paradigms,” no. March, 2024.

[33] A. El Yazidi, M. S. Azizi, Y. Benlachmi, and M. L. Hasnaoui, “Apache Hadoop-MapReduce on YARN framework latency,” *Procedia Comput. Sci.*, vol. 184, pp. 803–808, 2021.

[34] O. Azeroual and R. Fabre, “Processing big data with apache hadoop in the current challenging era of COVID-19,” *Big Data Cogn. Comput.*, vol. 5, no. 1, 2021.

[35] M. Y. Khalil and M. M. Hamad, “Big Data Management Using Hadoop,” *J. Phys. Conf. Ser.*, vol. 1804, no. 1, 2021.

[36] Y. Benlachmi and M. L. Hasnaoui, “Big data and spark: Comparison with hadoop,” *Proc. World Conf. Smart Trends Syst. Secur. Sustain. WS4 2020*, no. July 2020, pp. 811–817, 2020.

[37] A. Saxena, “A Survey on Big Data Analytics: Challenges, Open Research Issues and Tools,” *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 10, no. 12, pp. 1899–1908, 2022.

[38] T. Alam, “Cloud Computing and Its Role in the Information Technology,” *SSRN Electron. J.*, no. May, 2020.

[39] U. Gupta and R. Sharma, “‘Comparison of Different Cloud Computing Platforms for Data Analytics,’” *Lect. Notes Networks Syst.*, vol. 726 LNNS, no. September, pp. 67–78, 2023.

[40] Y. Zhao, K. Chen, H. Gao, and Y. Li, “Performance analysis of cloud resource allocation scheme with virtual machine inter-group asynchronous failure,” *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 36, no. 7, p. 102155, 2024.

[41] S. Samsam Shariat and B. Barekatain, “HATMOG: an enhanced hybrid task assignment algorithm based on AHP-TOPSIS and multi-objective genetic in cloud computing,” *Computing*, vol. 104, no. 5, pp. 1123–1154, 2022.

[42] Z. Mahmoodabadi and M. Nouri-Baygi, “An approximation algorithm for virtual machine placement in cloud data centers,” *J. Supercomput.*, vol. 80, no. 1, pp. 915–941, 2024.

[43] M. Guo, Q. Guan, W. Chen, F. Ji, and Z. Peng, “Delay-Optimal Scheduling

of VMs in a Queueing Cloud Computing System with Heterogeneous Workloads," *IEEE Trans. Serv. Comput.*, vol. 15, no. 1, pp. 110–123, 2022.

[44] S. Saeed, J. Olusegun, and E. Frank, "Batch processing with Apache Hadoop Mapreduce," no. November, 2024.

[45] R. Ghazali, S. Adabi, A. Rezaee, D. G. Down, and A. Movaghar, "CLQLMRS: improving cache locality in MapReduce job scheduling using Q-learning," *J. Cloud Comput.*, vol. 11, no. 1, 2022.

[46] P. Pandit, "Hadoop Ecosystem: Technology Study, Architecture and Analysis," *Researchgate.Net*, no. September, 2021.

[47] W. Alawad and A. Balobaid, "In big data era: Analysis of Hadoop cluster performance," *2021 Int. Conf. Women Data Sci. Taif Univ. WiDSTaif 2021*, 2021.

[48] I. Lebdaoui and G. Orhanou, "Preliminary Analysis of HDFS Read Operation, Threats Impacts and Mitigation," *Int. J. Eng. Trends Technol.*, vol. 72, no. 9, pp. 33–48, 2024.

[49] M. Elkawkagy and H. Elbeh, "High performance hadoop distributed file system," *Int. J. Networked Distrib. Comput.*, vol. 8, no. 3, pp. 119–123, 2020.

[50] W. I. Nemouchi, S. Boudouda, and N. E. Zarour, "A Dynamic Scaling Approach in Hadoop YARN," *Int. J. Organ. Collect. Intell.*, vol. 12, no. 2, pp. 1–17, 2021.

[51] G. Schmutz and G. Schmutz, "Real-Time Analytics with Apache Cassandra and Apache Spark," no. November, 2024.

[52] N. Ahmed, A. L. C. Barczak, M. A. Rashid, and T. Susnjak, "A parallelization model for performance characterization of Spark Big Data jobs on Hadoop clusters," *J. Big Data*, vol. 8, no. 1, 2021.

[53] F. Bajaber, S. Sakr, O. Batarfi, A. Altalhi, and A. Barnawi, "Benchmarking big data systems: A survey," *Comput. Commun.*, vol. 149, no. October 2019, pp. 241–251, 2020.

[54] J. Zhan, "A BenchCouncil view on benchmarking emerging and future computing," *BenchCouncil Trans. Benchmarks, Stand. Eval.*, vol. 2, no. 2, pp. 1–11, 2022.

[55] U. E. Ozdil and S. Ayvaz, “An Experimental and Comparative Benchmark Study Examining Resource Utilization in Managed Hadoop Context,” pp. 1–27, 2021.

[56] S. R. M. Zeebaree, “Distributed Systems for Data-Intensive Computing in Cloud Environments : A Review of Big Data Analytics and Data Management Indonesian Journal of Computer Science Distributed Systems for Data-Intensive Computing in Cloud Environments : A Review of Big Dat,” no. May, 2024.