Analisis Dataset TOP 1000 IMDb Movies Menggunakan Hadoop

Intan Permatahati, M. Nauval Perdana, Nanda Apriadi ,Tiara Putri Amanda, Zahra Maharani
Sistem Komputer, Universitas Sriwijaya, Indonesia

*Korespondensi: hatii3823@gmail.com

ARTICLE INFO

Article History:
- Received 10 January 2023
- Received in revised form 21 March 2023
- Accepted 10 April 2023
- Available online 30 July 2023

ABSTRAK

Dalam studi ini, kami mengeksplorasi analisis dataset TOP 1000 IMDb Movies dengan memanfaatkan keunggulan kerangka kerja Hadoop. Pertama-tama, untuk mengatasi tantangan pemrosesan data yang tidak terstruktur, kami menerapkan strategi khusus melalui infrastruktur yang mampu menangani volume besar dan kompleksitas data, yang umumnya dikenal sebagai Big Data. Kerangka kerja Hadoop telah terbukti efektif dalam menangani tugas-tugas semacam ini dengan menyediakan lingkungan yang terdistribusi untuk pemrosesan data secara paralel. Kemudian, untuk mengoptimalkan kueri data pada dataset IMDb, kami mengadopsi pendekatan dengan menggunakan Python. Keunggulan Python sebagai bahasa pemrograman untuk analisis data memberikan fleksibilitas dan keterbacaan kode, sementara implementasi multinode memungkinkan pemrosesan data yang terdistribusi untuk meningkatkan efisiensi. Gabungan dari kerangka kerja Hadoop, Python, dan pendekatan multinode menghasilkan pendekatan analisis yang efisien dan handal untuk dataset film IMDb TOP 1000.

Kata Kunci: IMDB , Top Movies, Hadoop, Big Data, Multinode

ABSTRACT

In this study, we explore the analysis of the TOP 1000 IMDb Movies dataset by leveraging the advantages of the Hadoop framework. Firstly, to address the challenges of processing unstructured data, we implement a specialized strategy through an infrastructure capable of handling large volumes and the complexity of data, commonly known as Big Data. The Hadoop framework has proven effective in handling such tasks by providing a distributed environment for parallel data processing. Subsequently, to optimize data queries on the IMDb dataset, we adopt an approach using Python. The superiority of Python as a programming language for data analysis provides flexibility and code readability, while the multinode implementation enables distributed data processing to enhance efficiency. The combination of the Hadoop framework, Python, and the multinode approach results in an efficient and reliable analytical approach for the TOP 1000 IMDb Movies dataset.

Keywords: IMDB, Top Movies, Hadoop, Big Data, Multinode

1. PENDAHULUAN

Dalam era revolusi industri 4.0, Big Data bukan lagi menjadi istilah asing. Terlampauinya kemampuan media penyimpanan maupun sistem database yang pertumbuhan datanya terus berlipat ganda dari waktu ke waktu menjadi alasan tercetusnya istilah big data. Big Data adalah istilah yang menggambarkan data yang berukuran besar, berkecepatan tinggi, kompleks, dan variabel data yang memerlukan teknik dan teknologi tingkat tinggi untuk memungkinkan dilakukannya pengumpulan, penyimpanan, distribusi, manajemen, dan analisis

informasi [1] . Big data berguna dalam memprediksi dan menganalisis penyebab suatu masalah pada sistem. Hal ini juga dapat meminimalisir terjadinya kegagalan, apalagi ditambah dengan hasil analisisnya langsung dapat ditampilkan. Selain itu, juga bisa menjadi referensi untuk pengembangan produk juga mengurangi waktu dan biaya. Hal ini yang menjadi alasan penerapan big data di era sekarang hampir di segala bidang. Dalam analisis kali ini, kami menggunakan Hadoop untuk mengolah big data [2] .

Teknologi Big Data di massa sekarang sudah banyak sekali membantu orang atau perusahaan. Di massa sekarang, Big Data memengaruhi banyak pihak dikarenakan ia adalah dasar empirik untuk banyak strategi pemasaran serta keputusan publik. Hal ini dikarenakanBig Data dapat melihat pola perilaku orang. Contoh pemanfaatan Big Data dalam dunia usaha yaitu membantu mendapatkan informasi menegnai feedback masyarakat ke produk-produk yang diluncurkan melalui analisis pendapat masyarakat di media sosial, dan juga membantu perusahaan untuk mengambil keputusan sehingga dapat menghasilkan keputusan terbaik berdasarkan Big Data. Selain membantu menganalisa bisnis, teknologi ini juga dapat digunakan secara luas ranah pemerintahan. Diantaranya, pemanfaatan Big Data di sektor publik dapat memberikan informasi mengenai feedback Masyarakat terhadap layanan pemerintah di masyarakat. Bahkan dapat membantu untuk menemukan solusi terhadap masalah yang ada dengan cara menganalisis masalah tersebut [3]

Dalam analisis ini, kami menggunakan Hadoop sebagai kerangka kerja pemrosesan big data untuk menganalisis dataset besar seperti IMDb Top 1000 Movies. Bertujuan untuk menambah wawasan mengenai rating di berbagai film, jika untuk keperluan pribadi hal ini bisa berguna untuk memutuskan film yang ingin ditonton. Namun, untuk industri film, hal ini dapat berguna untuk pengambilan keputusan di industri film. Mengetahui hubungan antara genre dan klasifikasi film untuk mengidentifikasi tren atau pola spesifik yang mungkin berguna bagi produser, sutradara, dan pemangku kepentingan lainnya di industri film ataupun hanya untuk sekedar pengetahuan umum. Untuk hasil analisis film yang lebih baik, dilakukannya penambangan data mendalam, lalu diolah menggunakan Hadoop, yang dapat digunakan sebagai landasan untuk pengembangan analisis lebih lanjut atau aplikasi bisnis di industri film. Eksperimen ini dapat memberikan wawasan tentang bagaimana rating tiap film yang masuk dalam kategori 1000 terbaik dan bagaimana Hadoop dapat digunakan untuk mengelola dan menganalisis kumpulan data besar secara efisien [4]. Penelitian ini berguna untuk membantu pengguna dalam pemilihan film yang cocok dengan preferensi mereka. Umumnya, pengguna seringkali merasa kebingungan dalam menemukan genre film yang disukai. Bahkan, terkadang untuk mencari film yang ingin dinikmati dapat menghabiskan waktu yang lama. Maka dari itu, analisis rating terhadap genre film ini harapannya akan membantu dalam memahami tren dan preferensi pengguna. Sehingga, tidak terjadinya ekspetasi yang tidak sesuai terhadap sebuah film. Pengurutan berdasarkan rating ini membantu menemukan film-film yang diakui oleh pengguna lain sebagai film yang bagus sesuai preferensi pengguna [5]. Big data adalah istilah yang digunakan untuk menggambarkan jumlah data yang besar, kecepatan dalam pengumpulannya, dan keragaman data yang diperlukan untuk mengatasi tantangan pengelolaan, pemrosesan, dan analisis data. Ada beberapa tools yang sangat mumpuni untuk mengelola dan menganialisi big data, diantaranya:

1. Apache Hadoop : Kerangka kerja sumber terbuka yang populer untuk menyimpan dan memproses big data. Ini termasuk Hadoop Distributed File System (HDFS) dan MapReduce yang memungkinkan analisis terdistribusi [6] .
2. Apache Spark : Kerangka kerja komputasi cepat yang memungkinkan analisis data real-time dan batch. Ini lebih cepat dari pada MapReduce karena menggunakan pemrosesan dalam memori [7] .

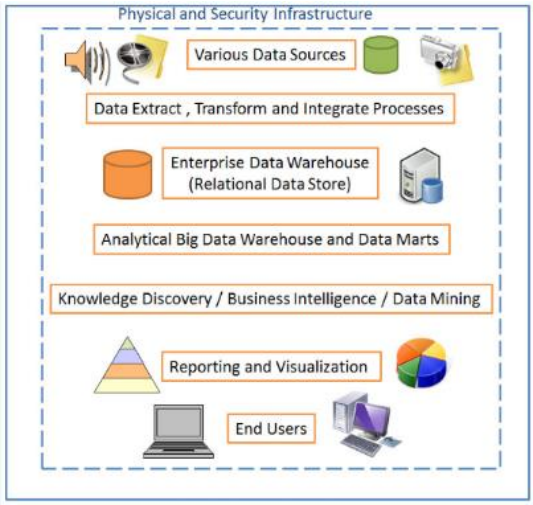
3. Apache Flink: Kerangka kerja pemrosesan aliran data yang sangat baik untuk aplikasi yang memerlukan pemrosessan aliran data waktu nyata.
4. Apache Kafka: Platform streaming data yang dapat digunakan untuk mengumpulkan, mengirim, dam menganalisis aliran data secara real-time [8] .
5. Apache HBase: Database NoSQL yang dirancang untuk menyimpan dan mengakses big data dengan cepat.
6. DataBase NoSQL: Beberapa databse NoSQL seperti MongoDB, Cassandra dan Couchbase cocok untuk menyimpan dan memproses data semi-terstruktur dan tidak terstruktur [9] .
7. SQL-on-Hadoop: Kerangka kerja seperti Apache Hive dan Apache Impala mengaktifkan kueri SQL terhadap data yang disimpan di Hadoop [10].

2. TINJAUAN PUSTAKA

2.1 Big Data

Kata ini berarti kumpulan data yang besar dan terus berkembang, termasuk data heterogen, tidak terorganisisr, dan semi-terorganisir. Big data bersifat kompleks sehingga membutuhkan teknologi dan algoritma canggih. Big data didefinisikan menjadi 3 karakteristik utama [11] .

- Volume:
  Data digital dalam jumlah besar terus dihasilkan dari berbagai hal (TIK, ponsel cerdas, kode produk, jejaring sosial, sensor, log, dll. Intinya, volume berpacu pada data yang terkumpul, tentunya hal ini meliputi triliunan atau lebih baris data.
- Kecepatan:
  Melihat dari seberapa cepat pertumbuhan data baru yang harus diolah.
- Varian:
  Menggambarkan keberagaman sumber dan tipe data. Salah satu tantangan mengolah big data yaitu kita harus mampu mengelola dan menganalisis data yang ada.

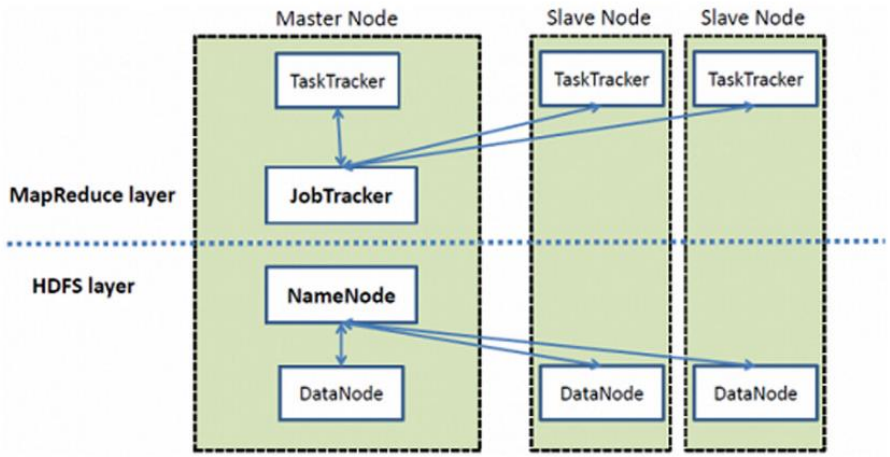


Gambar 1. Arsitektur Big Data

Big Data Analytics membantu mengidentifikasi transformator yang berisiko dan mendeteksi perilaku tidak biasa dari perangkat yang terhubung. Dengan cara ini, Grid Utilities dapat memilih tindakan terbaik. Analisis real-time dari Big Data yang diperoleh membantu memodelkan skenario kegagalan.

2.2 Hadoop

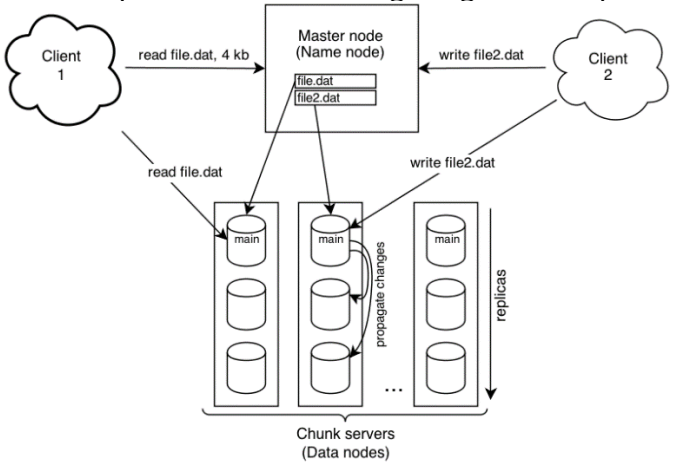
Big Data saat ini menjadi tren teknologi yang signifikan. Salah satu platform perangkat lunak yang dapat digunakan untuk mengelola big data adalah Hadoop. Secara umum, Hadoop adalah perangkat lunak yang memungkinkan penggabungan satu hingga banyak komputer agar dapat bekerja sama dan terhubung untuk menyimpan serta mengelola data secara terpadu [12]. Biasanya Hadoop menggunakan model pemrograman Hive atau MapReduce untuk menyimpan dan memproses Big Data.



Gambar 2. Arsitektur Hadoop

2.3 Hadoop Distributed File System (HDFS)

Hadoop Distributed File System adalah salah satu pengembangan dari Apache, yaitu subtask dari Apache Hadoop. HDFS ini dikembangkan berdasarkan konsep dari GFS (Google File System). Pada Gambar 2 terdapat cara kerja HDFS, yaitu membagi file yang besar menjadi sejumlah bagian kecil sehingga membentuk cluster yang memungkinkan terjadinya pemrosesan secara paralel. Adapun komponen utama dari HDFS yaitu NameNode dan DataNode. NameNode bertugas menjadi master yaitu berkewajiban menyimpan informasi mengenai lokasi penempatan blok data dalam Hadoop Cluster. Sedangkan DataNode bertugas menjadi slave, yaitu mengamankan blok data yang diberikan kepadanya, serta melaporkan kondisinya secara berkala kepada NameNode. Untuk menyimpan data ke dalam HDFS kita perlu memastikan bahwa komputer sudah terhubung dengan Hadoop Cluster [13].

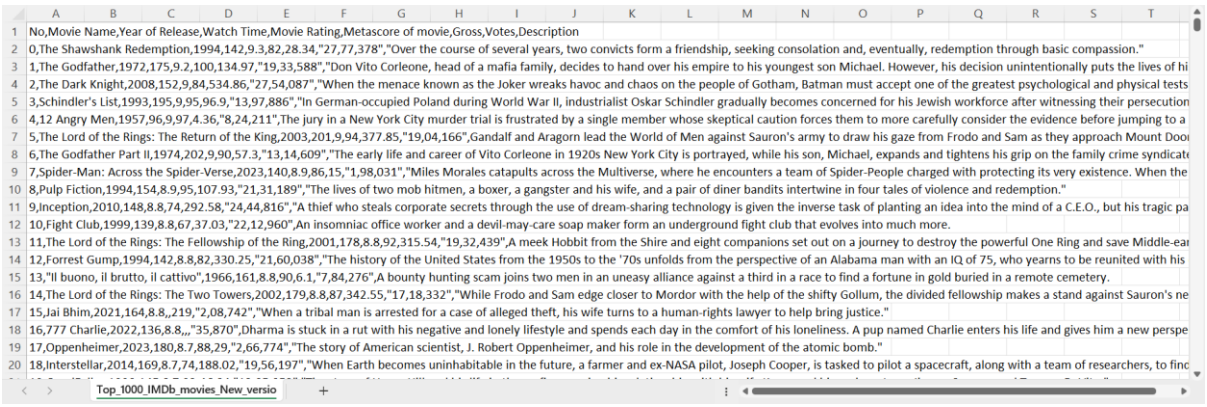


Gambar 3. Arsitektur HDFS

2.4 Dataset

Dataset adalah sekumpulan data. Kumpulan data ini biasanya direpresentasikan dalam bentuk tabel yang dimana setiap baris dan kolom menggambarkan variabel tertentu. Data tersebut diperoleh melalui pengukuran, pengamatan, studi atau analisis. Semakin akurat data yang dikumpulkan, analisis yang dibuat berdasarkan informasi tersebut lebih dapat dipercaya [14] . Terdapat dua jenis dataset :

1. Dataset Privat
   Dataset privat merujuk pada kumpulan data yang dapat diambil dari organisasi yang menjadi fokus penelitian. Contohnya mencakup instansi, rumah sakit, dan perusahan jasa [15] .
2. Dataset Publik
   Dataset publik merujuk pada kumpulan data yang diambil dari lembaga publik yang telah mendapatkan persetujuan dari para peneliti [16] .



Gambar 4. Dataset TOP 1000 IMDb Movies

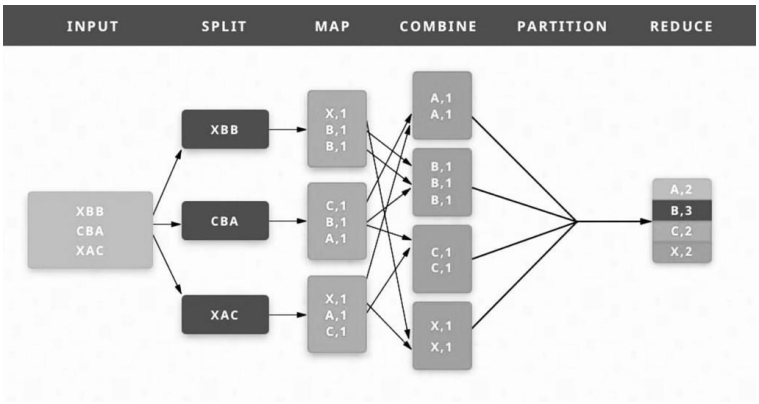
Pada penelitian ini, kami mengambil data melalui Kaggle. Kaggle merupakan platform yang memungkinkan untuk mengeksplorasi berbagai sumber data yang luas dan beragam. Seperti pada Gambar 3. Kami menggunakan dataset TOP 1000 IMDb Movies. Data tersebut merupakan data mentah yang belum diolah menggunakan Hadoop.

2.5 Python

Python merupakan bahasa pemrograman tingkat tinggi yang popular. Python dibuat oleh Guido Van Rossum pada awal 1990-an. Python menyedaikan perpustakaan yang komprehensif, memungkinkan pengembang untuk membuat aplikasi canggih dengan menggunakan source code yang tampak sederhana [17] . Python juga dapat dikatakan sebagai bahasa pemrograman yang dapat menyatukan formula kode yang jelas dan juga memiliki sebuah fungsionalitas Pustaka standar yang cukup banyak dan bersifat menyeluruh [18] . Dalam penelitian ini, Python akan menjadi bahasa pemrograman yang digunakan dalam pengurutan 1000 film teratas .

2.6 MapReduce

Metode pengolahan data yang dikenal sebagai MapReduce, dilakukan dengan cara membagi data menjadi potongan-potongan kecil, kemudian hasilnya disatukan kembali. MapReduce dimanfaatkan untuk mengolah data besar yang tersimpan dalam sistem file Hadoop (HDFS) [19] . Gambar 4 di bawah merupakan gambaran dalam pelaksanaan MapReduce yaitu tahap Map dan Reduce. Tahap Map berperan dalam membaca input dalam format pasangan Key/Value. Hasilnya akan dibaca dalam format pasangan Key/Value, kemudian dikelompokkan berdasarkan Key yang sama, proses ini disebut sebagai tahap Reduce [20] .



Gambar 5. Arsitektur MapReduce

3. METODELOGI PENELITIAN

3.1. Program yang Digunakan

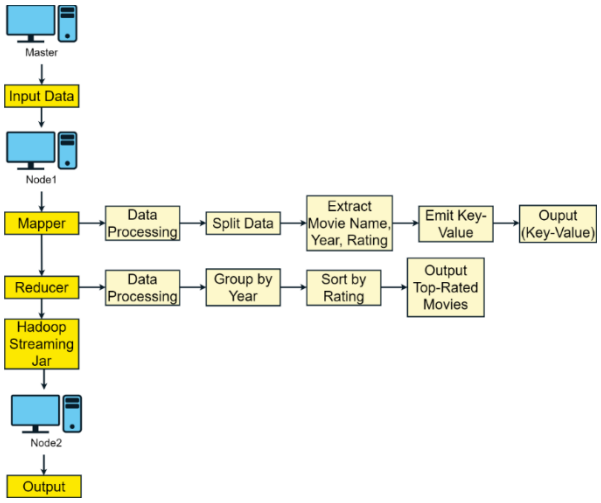
Dalam menganalisis dataset IMDb top 1000 movies ini kami menggunakan dua komponen utama, yakni Apache Hadoop dan bahasa pemrograman Python. Apache Hadoop berfungsi sebagai kerangka kerja pengolahan data besar yang menyediakan sistem penyimpanan terdistribusi melalui Hadoop Distributed File System (HDFS) dan proses pemrosesan data melalui MapReduce. Dalam melakukan MapReduce, Python digunakan untuk menulis kode Mapper dan Reducer, yang mana bertanggung jawab untuk memproses data input dan menghasilkan output.

3.2. Data yang digunakan pada program

Data yang kami gunakan dalam program ini diperoleh melalui sumber yang dapat diandalkan, yaitu https://www.kaggle.com/datasets/inductiveanks/top-1000-imdb-movies-dataset . Dataset ini merupakan referensi utama kami dalam analisis ini. Dataset ini mencakup beragam atribut terkait film seperti judul, tahun rilis, durasi, rating IMDb, dan informasi lainnya yang relevan. Sumber data ini dipilih karena keberagaman informasi yang tersedia dan relevansinya dengan tujuan analisis kami. Dengan menggunakan data ini, kami bertujuan untuk melakukan analisis statistik sesuai dengan konteks program yang kami lakukan.

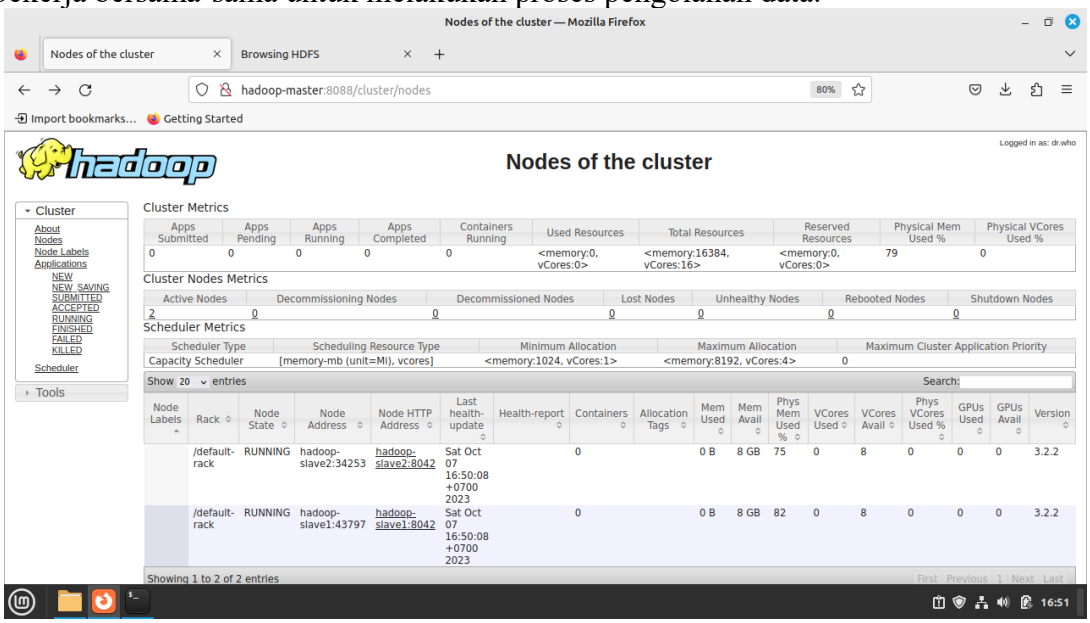
3.3 Cara kerja program

- Proses Data



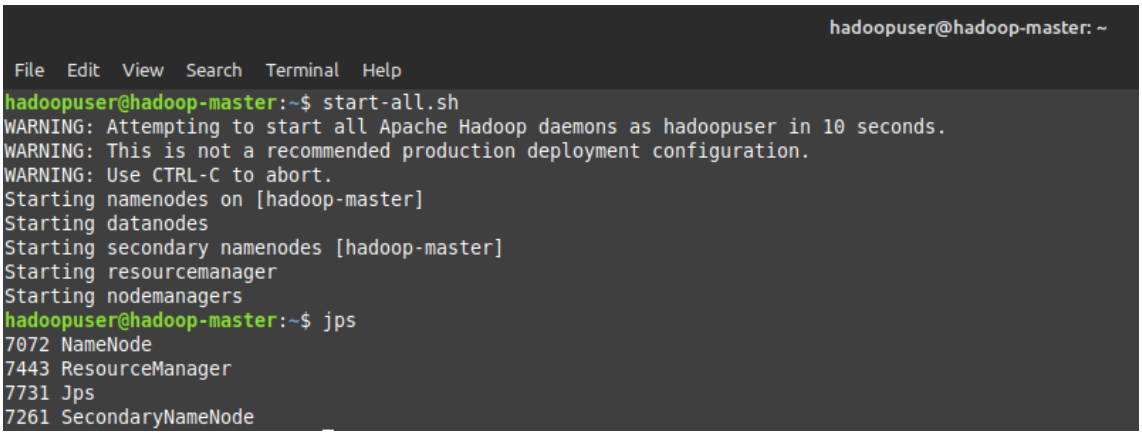
Gambar 6. Flowchart Proses Data

- Multinode Cluster
  Dalam menjalankan proses pengolahan data ini, kami memanfaatkan sebuah multinode cluster yang terdiri dari tiga node, yaitu master, slave1, dan slave2. Ketiga node ini bekerja bersama-sama untuk melakukan proses pengolahan data.

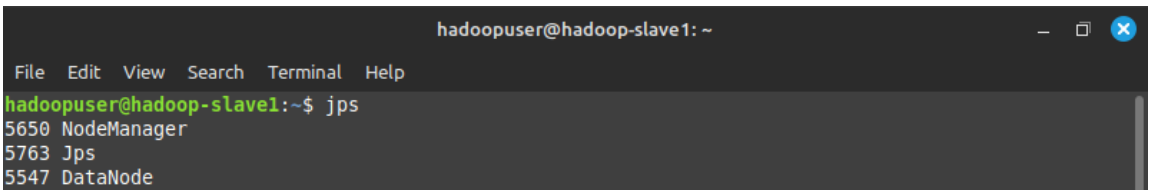


Gambar 7. Multinode Cluster

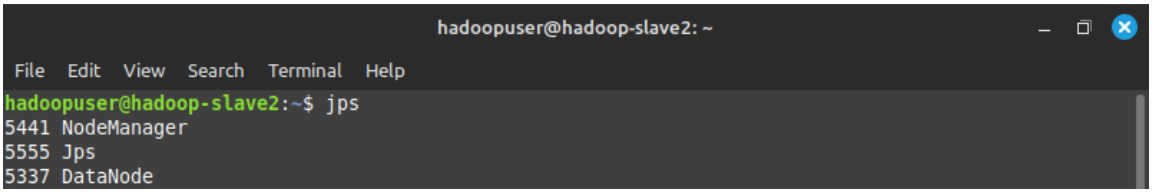
- Memulai Hadoop
  Buka terminal lalu ketik perintah ‘start-all.sh’, kita dapat melihat status dari semua proses java yang sedang berjalan pada setiap node dengan perintah ‘jps’.



Gambar 8. Memulai Hadoop & JPS Master



Gambar 9. JPS Slave1



Gambar 10. JPS Slave2

- Penginputan Data
  Pertama, buatlah sebuah direktori untuk menyimpan data.

```
hadoopuser@hadoop-master:~$ hadoop fs -mkdir /data
```

  Lalu input data dari local system node master ke direktori Hadoop yang baru dibuat dengan perintah '-copyFromLocal'.

```
hadoopuser@hadoop-master:~$ hadoop fs -copyFromLocal /home/hadoopuser/data/Top_1000_IMDb_movies_New_version.csv /data
```

  Kita dapat melihat bahwa data tersebut sudah diinput dengan perintah '–ls /data'

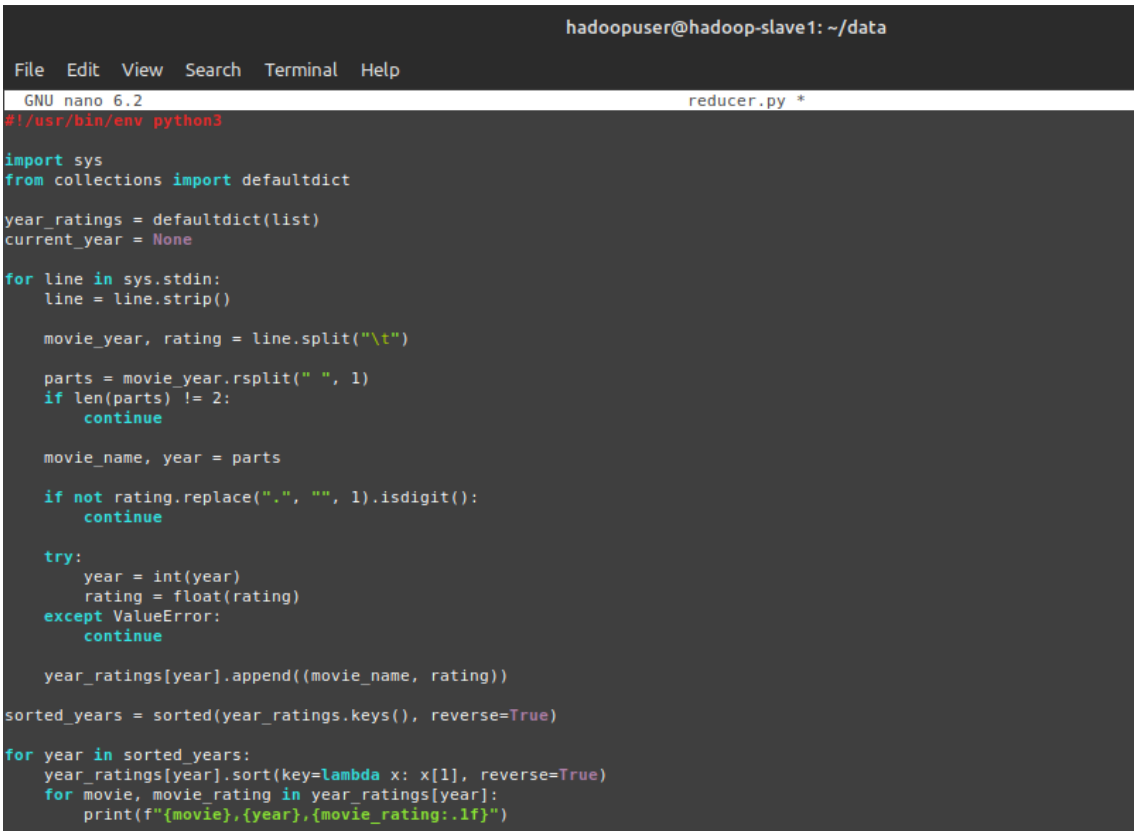
```
hadoopuser@hadoop-master:~$ hadoop fs -ls /data
Found 1 items
-rw-r--r--   2 hadoopuser supergroup     204174 2023-10-07 17:02 /data/Top_1000_IMDb_movies_New_version.csv
```

- Proses Mapreduce
  Proses mapreduce kami lakukan pada node slave1 dengan tiga proses yaitu membuat kode mapper dan reducer dengan bahasa pemrograman python dan menggunakan tools Hadoop Streaming Jar untuk menjalankan tugas pemrosesan pada kode mapper dan reducer.

  - Mapper

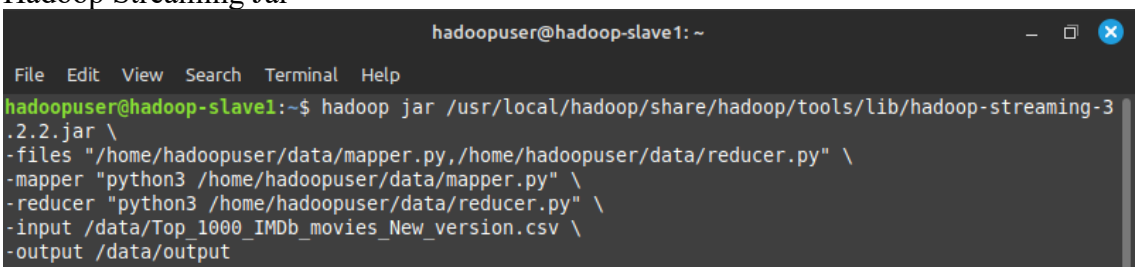
Gambar 11. Proses Mapper

- Reducer



Gambar 12. Proses Reducer

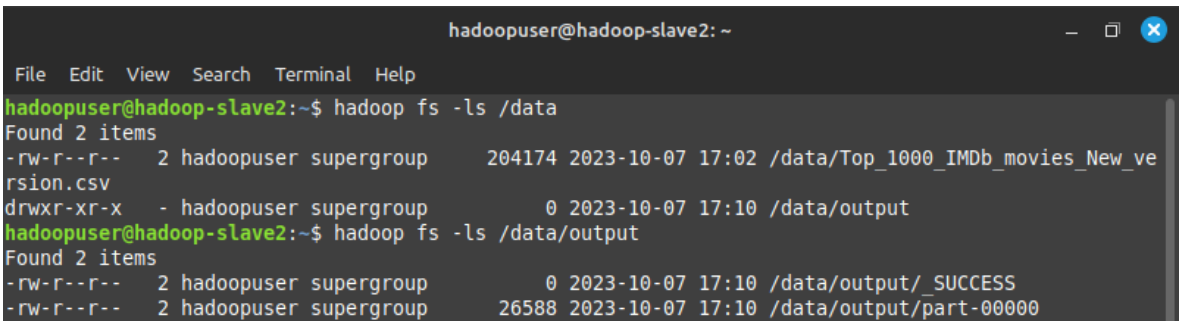
- Hadoop Streaming Jar



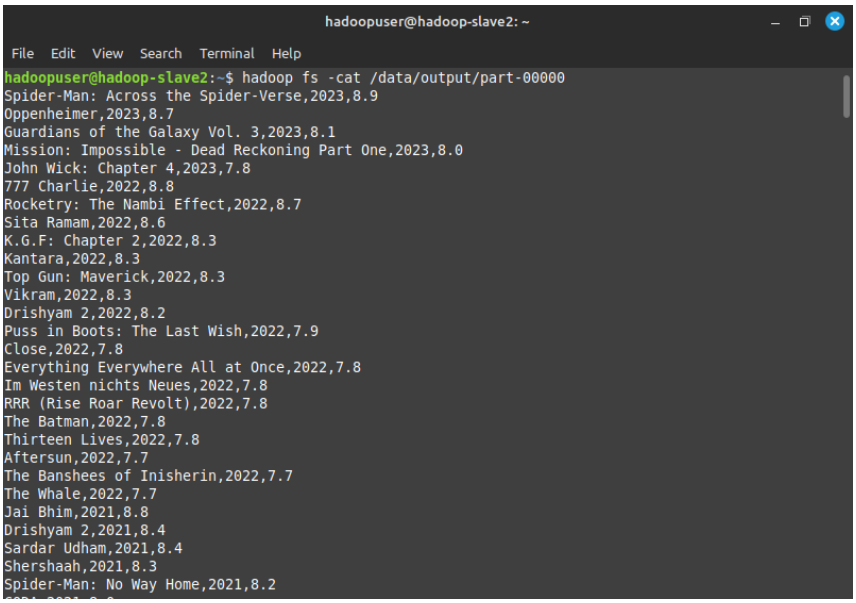
Gambar 13. Hadoop Streaming Jar

- Output

Setelah proses mapreduce berhasil dijalankan maka akan ada dua file yang muncul pada direktori Hadoop yang telah ditentukan untuk menyimpan output, yaitu file bernama _SUCCES yang menandakan bahwa proses mapreduce telah berhasil selesai tanpa kesalahan dan file part-xxxxx yang merupakan output yang dihasilkan oleh mapreduce. Kami menggunakan node slave2 untuk melihat output, menggunakan perintah ‘hadoop fs -cat /data/output/part-00000’ untuk melihat isi dari file tersebut.



Gambar 14. Direktori Output

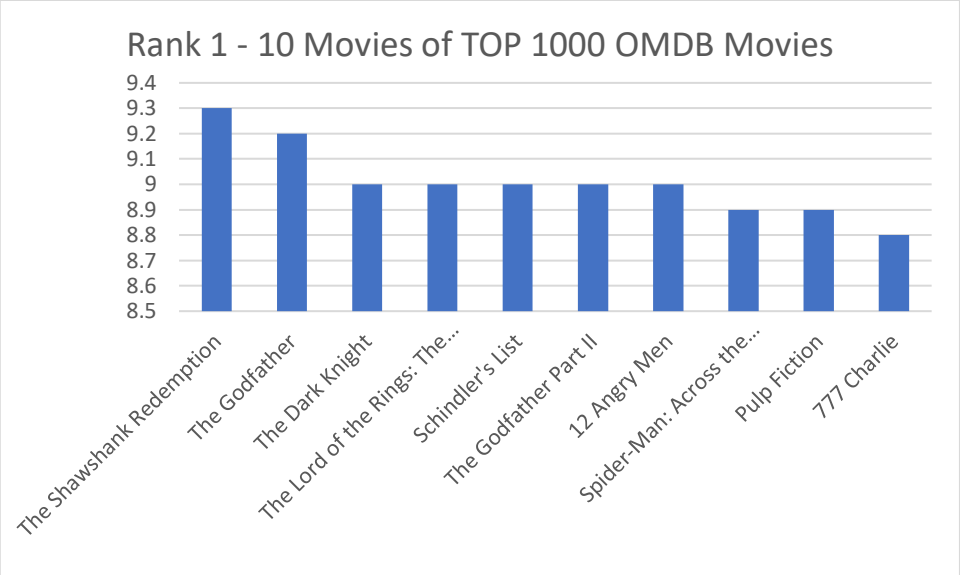


Gambar 15. Hasil Tampilan Output

4. HASIL PENELITIAN

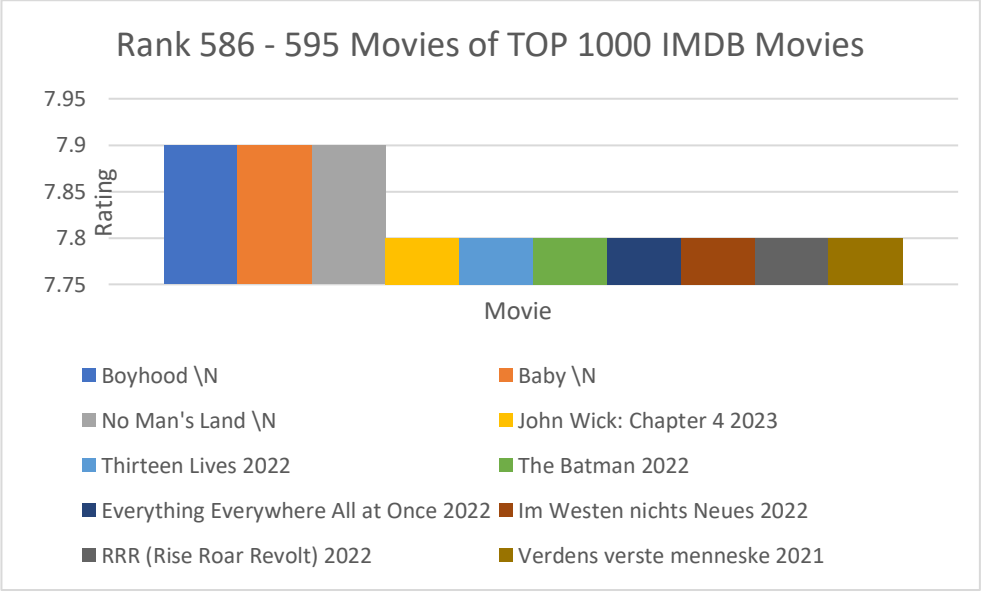
Berdasarkan pengujian yang dilakukan dengan menginput file data yang belum terstruktur ke dalam Apache Hadoop. Isi data berjumlah 1000 dengan besar 179 Kb menghasilkan hasil akhir seperti pada gambar 6. Kami merepresentasikan hasil data tersebut pada chart di bawah ini.

- Diagram batang di bawah menunjukkan 10 film tertinggi dari total keseluruhan 1000 data.



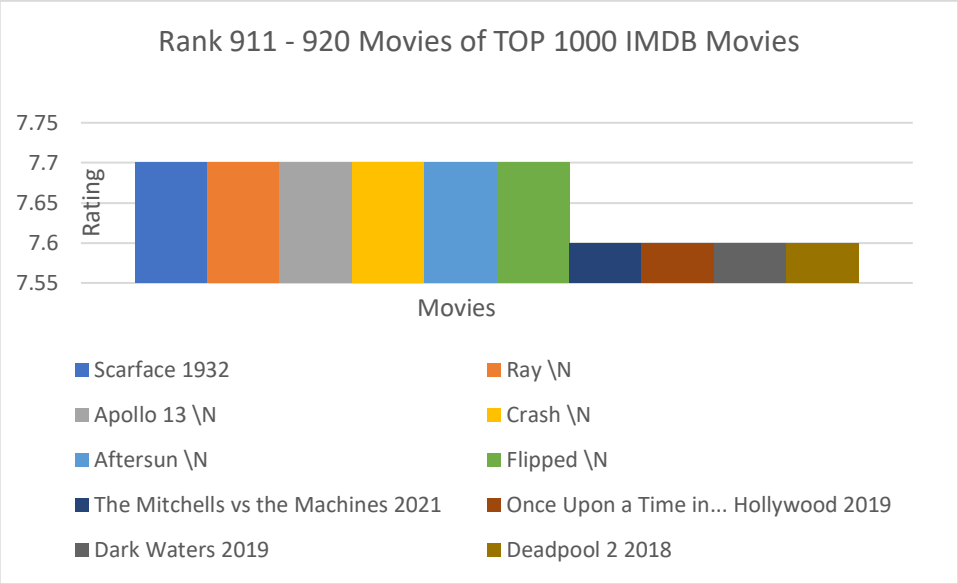
Gambar 16. Top Movie

- Diagram batang di bawah menunjukkan peringkat film dari 586 - 595 dari total keseluruhan 1000 data.



Gambar 17. Rangking 586-595

- Diagram batang di bawah menunjukkan peringkat film dari 911 - 920 dari total keseluruhan 1000 data.



Gambar 18. Rangking 911-920

5. KESIMPULAN

Apache Hadoop sebagai kerangka kerja pengolahan data besar yang menyediakan sistem penyimpanan terdistribusi melalui Hadoop Distributed File System (HDFS) dan model pemrograman MapReduce untuk pemrosesan secara paralel. Dalam melakukan MapReduce, Python digunakan untuk menulis kode Mapper dan Reducer, yang bertanggung jawab untuk memproses data input dan menghasilkan output. Hasil penelitian ini berfokus pada rekomendasi film, memberikan manfaat kepada pengguna untuk menemukan rekomendasi film berdasarkan urutan ratingnya.

Dari 1000 data yang dianalisis. Ternyata, setelah diolah, terdapat beberapa data yang rusak, kerusakannya bisa terletak pada tahun terbit film ataupun ratingnya. Sehingga data yang diambil untuk divisualisasikan, minimal hanyalah yang tidak ada tahunnya. Kami memvisualisasikan pengelompokkan 1000 data tersebut kedalam 3 klaster, yaitu klister 1 (rank 1 – 10), klister 2 (rank 586 – 595), dan klister 3 (rank 911 – 920). Untuk klaster pertama rating tertinggi diraih oleh film "The Shawshank Redemption – 1994" dengan rating 9,3. Sedangkan di klaster kedua ada film "No Man's Land" dengan rating 7,9 dan "John Wick 4 – 2023" dengan rating 7,8. Dan untuk klaster terakhir ada film "Scarface – 1932" dengan rating 7,7 dan "Deadpool 2 – 2018" dengan rating 7,6.

DAFTAR PUSTAKA

[1] N. U. Shahid and N. J. Sheikh, "Impact of Big Data on Innovation, Competitive Advantage, Productivity, and Decision Making: Literature Review," *Open Journal of Business and Management*, vol. 09, no. 02, pp. 586–617, 2021, doi: 10.4236/ojbm.2021.92032.

[2] E. Riyandani, "'Big Data vs Big Information vs Big Knowledge' Oleh: Imam Cholissodin," 2016. [Online]. Available: http://bit.ly/2x8ta9S
[3] M. Febriansyah Trisnadi, S. Al Faraby, and M. Dwifebri, "Sentiment Analysis pada Movie Review Menggunakan Feature Selection Mutual Information dan K-Nearest Neighbour Classifier."
[4] M. Febriansyah Trisnadi, S. Al Faraby, and M. Dwifebri, "Sentiment Analysis pada Movie Review Menggunakan Feature Selection Mutual Information dan K-Nearest Neighbour Classifier."
[5] O. Azeroual and R. Fabre, "Processing Big Data with Apache Hadoop in the Current Challenging Era of COVID-19," *Big Data and Cognitive Computing*, vol. 5, no. 1, p. 12, Mar. 2021, doi: 10.3390/bdcc5010012.
[6] G. Ravichandran, "Big Data Processing with Hadoop : A Review," *International Research Journal of Engineering and Technology*, 2017, [Online]. Available: www.irjet.net
[7] A. Ghaffar Shoro and T. Rahim Soomro, "Big Data Analysis: Apache Spark Perspective," 2015. [Online]. Available: https://www.researchgate.net/publication/272825265
[8] S. F. Astika, M. Jauhari, N. Isbatuzzin, M. Salman, and K. Ramli, "BUILDING A DYNAMIC SCALABLE PARALLEL CLOUD-BASED SNORT NIDS USING CONTAINERS AND BIG DATA," *Journal of Southwest Jiaotong University*, vol. 56, no. 5, pp. 317–326, Oct. 2021, doi: 10.35741/issn.0258-2724.56.5.27.
[9] M. Riza *et al.*, "Relational Data Modeling on the Document-Based NoSQL," 2022. [Online]. Available: https://www.researchgate.net/publication/367558756
[10] E. R. E. Sirait, "IMPLEMENTASI TEKNOLOGI BIG DATA DI LEMBAGA PEMERINTAHAN INDONESIA," *Jurnal Penelitian Pos dan informatika*, vol. 6, no. 2, p. 113, Dec. 2016, doi: 10.17933/jppi.2016.060201.
[11] S. Oliviandi, A. B. Osmond, and R. Latuconsina, "IMPLEMENTASI APACHE SPARK PADA BIG DATA BERBASIS HADOOP DISTRIBUTED FILE SYSTEM IMPLEMENTATION APACHE SPARK ON BIG DATA BASED HADOOP DISTRIBUTED FILE SYSTEM."
[12] Y. Surahman and H. Saptono, "Implementasi dan Analisis Kinerja HDFS sebagai Infrastruktur Pembangunan Big Data," *Jurnal Informatika Terpadu*, vol. 4, no. 2, pp. 63–70, Sep. 2018, doi: 10.54914/jit.v4i2.159.
[13] Y. Surahman and H. Saptono, "Implementasi dan Analisis Kinerja HDFS sebagai Infrastruktur Pembangunan Big Data," *Jurnal Informatika Terpadu*, vol. 4, no. 2, pp. 63–70, Sep. 2018, doi: 10.54914/jit.v4i2.159.
[14] A. M. T I Sambi Ua *et al.*, "Penggunaan Bahasa Pemrograman Python Dalam Analisis Faktor Penyebab Kanker Paru-Paru Universitas Bina Nusantara," *Jurnal Publikasi Teknik Informatika (JUPTI)*, vol. 2, no. 2, 2023, doi: 10.55606/jupti.v2i2.1742.
[15] A. Yunita, H. B. Santoso, and Z. A. Hasibuan, "'Everything is data': towards one big data ecosystem using multiple sources of data on higher education in Indonesia," *J Big Data*, vol. 9, no. 1, Dec. 2022, doi: 10.1186/s40537-022-00639-7.
[16] A. M. T I Sambi Ua *et al.*, "Penggunaan Bahasa Pemrograman Python Dalam Analisis Faktor Penyebab Kanker Paru-Paru Universitas Bina Nusantara," *Jurnal Publikasi Teknik Informatika (JUPTI)*, vol. 2, no. 2, 2023, doi: 10.55606/jupti.v2i2.1742.
[17] C. Wibawa, S. Wirawan, M. Mustikasari, and D. T. Anggraeni, "KOMPARASI KECEPATAN HADOOP MAPREDUCE DAN APACHE SPARK DALAM

MENGOLAH DATA TEKS," *Jurnal Ilmiah Matrik*, vol. 24, no. 1, pp. 10–20, Apr. 2022, doi: 10.33557/jurnalmatrik.v24i1.1649.

[18] C. Wibawa, S. Wirawan, M. Mustikasari, and D. T. Anggraeni, "KOMPARASI KECEPATAN HADOOP MAPREDUCE DAN APACHE SPARK DALAM MENGOLAH DATA TEKS," *Jurnal Ilmiah Matrik*, vol. 24, no. 1, pp. 10–20, Apr. 2022, doi: 10.33557/jurnalmatrik.v24i1.1649.

[19] Y. Surahman and H. Saptono, "Implementasi dan Analisis Kinerja HDFS sebagai Infrastruktur Pembangunan Big Data," *Jurnal Informatika Terpadu*, vol. 4, no. 2, pp. 63–70, Sep. 2018, doi: 10.54914/jit.v4i2.159.

[20] C. Wibawa, S. Wirawan, M. Mustikasari, and D. T. Anggraeni, "KOMPARASI KECEPATAN HADOOP MAPREDUCE DAN APACHE SPARK DALAM MENGOLAH DATA TEKS," *Jurnal Ilmiah Matrik*, vol. 24, no. 1, pp. 10–20, Apr. 2022, doi: 10.33557/jurnalmatrik.v24i1.1649.