Pengolahan Data *Polling* berbasis Media Sosial menggunakan *MapReduce* pada *Framework Hadoop*

Yusuf Yunadian[1], Dr. Hilal H Nuha, S.T., M.T.[2], Sidik Prabowo, S.T., M.T.
[1,2,3]Fakultas Informatika, Universitas Telkom, Bandung
[1]yyunadian@student.telkomuniversity.ac.id [2]hilalnuha@gmail.com
[3]sidikprabowo@telkomuniversity.ac.id

Abstrak

Pengolahan data dari sebuah sistem *Polling* menjadi hal yang sangat penting karena data hasil olah tersebut dapat digunakan oleh publik untuk dijadikan acuan di dalam menghadapi sebuah persoalan yang berkembang di masayarakat. Pertumbuhan di dalam penggunaan media sosial semakin meningkat dari tahun ke tahun, di mana Indonesia sendiri merupakan pengguna media social *Twitter* terbanyak ke-5 di dunia. Di dalam proses pengolahan data, jika data yang akan diolah berukuran cukup besar, akan memperlambat komputasi yang dilakukan. Hal tersebut mendorong penulis untuk membuat sebuah sistem yang dapat memperoses data *polling* yang dilakukan melalui media sosial dengan waktu yang lebih efisien. *Hadoop* merupakan salah satu sistem yang optimal untuk digunakan di dalam pengolahan data *polling* pada *Tugas* Akhir ini. Pada *Hadoop* terdapat 2 modul utama yaitu *Hadoop Distributed File System* (HDFS) yang merupakan sistem penyimpanan terdistribusi, dan *MapReduce* yang merupakan algoritma/komputasi pada *Hadoop*. Pada pengoloahan data *polling* ini menggunakan program *wordcount* dengan *MapReduce* pada *Hadoop* dan dengan program *wordcount* tanpa *MapReduce*. Dilakukan pengujian terhadap 2 metode tersebut, dengan diujikan menggunakan beberapa data dengan ukuran dari yang kecil sampai ke yang berukuran besar. Dan dihasilkan bahwa, *MapReduce* lebih unggul dalam segi kecepatan proses data dibandingkan dengan metode proses data tanpa *MapReduce.* Dengan rata-rata dari data yang diujikan, menggunakan *MapReduce* pada *Hadoop* dapat memproses data 1,3 kali lebih cepat dibandingkan tanpa *MapReduce* pada *Hadoop*.

Kata Kunci: *polling, Hadoop, MapReduce, wordcount,* kecepatan proses.

Abstract

Data processing from a Polling system becomes very important because the results of data processing can be used by the public to be used as a reference in dealing with a problem that is developing in the community. Growth in the use of social media has increased from year to year, where Indonesia itself is the fifth largest user of Twitter social media in the world. In the data processing, if the data to be processed is large enough, it will slow down the computation done. This encourages the author to create a system that can process polling data conducted through social media with a more efficient time. Hadoop is one of the optimal systems for use in polling data processing in this Final Project. In Hadoop there are 2 main modules namely Hadoop Distributed File System (HDFS) which is a distributed storage system, and MapReduce which is an algorithm / computation on Hadoop. In processing this poll data using the wordcount program with MapReduce on Hadoop and with the wordcount program without MapReduce. Tests of the 2 methods were conducted, tested using some data with sizes from small to large. And it is produced that, MapReduce is superior in terms of data processing speed compared to the data processing method without MapReduce. With an average of data tested, using MapReduce on Hadoop can process data 1.3 times faster than without MapReduce on Hadoop.

Keyword: polling, Hadoop, MapReduce, wordcount, processing speed.

1. Pendahuluan

1.1 Latar Belakang

Pengguna media sosial di Indonesia semakin bertambah seiring dengan banyaknya pengguna *smartphone* (telepon pintar). Fungsi media sosial saat ini bukan hanya sekedar untuk berkomunikasi, berbagi informasi dan eksistensi. Masyarakat pun dapat melakukan jajak pendapat (*polling*) di media sosial. Metode *polling*

menjadi metode yang paling efektif dalam mendapatkan informasi yang cepat mengenai masalah atau isu yang berkembang di masyarakat. *Polling* digunakan untuk mendapatkan informasi tentang suatu fenomena, dalam hal ini yang ingin didapat dari *polling* adalah sikap, pandangan, dan keyakinan masyarakat terhadap isu-isu yang berkembang [1]. Karena itu dapat juga dikatakan bahwa *polling* adalah penerapan praktis dari metode survei, pemakaian metode survei untuk mengukur pendapat publik seperti isu-isu politik. Media Sosial merupakan *platform* komunikasi yang memiliki kemampuan untuk mengetahui opini publik terhadap suatu isu secara cepat [1].

Biasanya untuk melakukan sebuah *polling* kita butuh setidaknya data yang cukup besar, untuk mendapatkan hasil yang maksimal. Data yang besar ini sering disebut juga dengan, istilah *Big Data*. *Big Data* merupakan kumpulan data yang besar, sangat variatif, dan mungkin tidak terstruktur [2]. *Big Data* sangatlah besar sehingga sulit untuk prosedur konvensional di dalam menganalisa *big data*. Dengan data yang besar, dapat menjadikan analisa terhadap suatu fenomena lebih sempurna, dan jika berhasil menganalisa sebuah data tersebut akan membantu di dalam pengambilan sebuah keputusan dengan lebih baik [2]. Yang diharapkan juga data hasil analisa bisa efisien mewakili suara yang didapat, kost lebih murah, dan kecepatan di dalam pemrosesan. Sehingga diperlukan algoritma khusus sehingga informasi yang mendalam mudah didapatkan dan membantu di dalam pengambilan keputusan yang lebih baik.

Hadoop merupakan salah satu sistem terdistribusi yang diperuntukan untuk memproses data berukuran besar [3]. *Hadoop* merupakan sebuah *framework* untuk penyimpanan dan pemprosesan data skala besar yang terdiri dari beberapa modul, yaitu *Hadoop Common*, *Hadoop Distributed File System* (HDFS), *Hadoop* YARN, dan *Hadoop MapReduce* [4].

Tugas Akhir ini mengimplementasikan metode *MapReduce* pada *Framework Hadoop* yang digunakan untuk memproses data *polling* supaya lebih efisien dalam segi waktu proses data. Data yang akan diproses berasal dari *web* (happypoll.id), dimana untuk responden yang akan melakukan *polling* diharuskan untuk *login* terlebih dahulu. Hal tersebut untuk mendapatkan informasi, diantaranya alamat, *username*, *email*, dan informasi akun *twitter* responden lainnya. Pengolahan data memanfaatkan komputasi *MapReduce* dengan program *wordcount*, yang mana dilakukan beberapa kali pengujian. Yaitu pengolahan data *polling* dengan program *wordcount* dengan *MapReduce* pada *Hadoop* dan program *wordcount* tanpa *MapReduce* atau program *wordcount* biasa dengan bahasa java. Hal ini dilakukan untuk mengetahui seberapa efektif jika *Hadoop* digunakan untuk proses data *polling* berbasis media sosial pada Tugas Akhir ini.

1.2 Perumusan Masalah

Berdasarkan latar belakang yang telah diuraikan di atas, terdapat beberapa permasalahan yang dibahas dalam Tugas Akhir ini, yaitu sebagai berikut:

1. Bagaimana merancang sistem untuk pengolahan data *polling* berbasis media sosial dengan memanfaatkan *MapReduce* pada *Hadoop*?
2. Bagaimana program implementasi program *wordcount* dengan *MapReduce* pada *Hadoop* untuk proses data *polling*?
3. Bagaimana kecepatan proses data *wordcount* dengan *MapReduce* dan *wordcount* tanpa *MapReduce*?
4. Apakah *Block* dapat mempengaruhi proses data pada *Hadoop* yang di-*setting* dengan *Multi Node Cluster*?

1.3 Batasan Masalah

Batasan masalah dalam Tugas Akhir ini adalah:

1. Data yang digunakan merupakan data hasil *polling* di *web* happypoll.id, dimana *voters* harus *login* terlebih dahulu melalui *twitter*.
2. Data *polling* berbentuk file .text yang disesuaikan.

3. Pengolahan data memanfaatkan komputasi *MapReduce* pada *Framework Hadoop* dengan program *wordcount*.
4. Dilakukan pengujian untuk membandingkan program *wordcount* dengan *MapReduce* pada *Hadoop* dan program *wordcount* tanpa *Hadoop*.
5. Dilakukan pengujian pada *Hadoop* dengan ukuran *block* yang diperbesar.

1.4 Tujuan

Adapun tujuan dari Tugas Akhir ini, yaitu sebagai berikut:

1. Membuat sistem pengolahan data *polling* berbasis media sosial, dengan memanfaatkan *MapReduce* pada *framework Hadoop*.
2. Menjalankan aplikasi *wordcount* yang berjalan dengan komputasi *MapReduce* pada *Framework Hadoop* yang dikonfigurasi secara *MultiNode Cluster*.
3. Melakukan pengujian program *wordcount* dengan *MapReduce Hadoop* dan tanpa *MapReduce Hadoop*, untuk mengetahui metode mana yang lebih unggul dari segi kecepatan proses data.
4. Melakukan pengujian pada program *wordcount* dengan *MapReduce Hadoop*, dengan memperbesar ukuran *block* untuk mengetahui pengaruh dari *block*.

1.5 Organisasi Penulisan

Penelitian TA ini disusun dengan struktur sebagai berikut: Bagian pertama berupa pendahuluan yang berisi mengenai latar belakang, perumusan masalah, batasan masalah, tujuan, dan organisasi penulisan. Bagian kedua berisi mengenai studi terkait, referensi-referensi yang berkaitan dengan TA penulis. Bagian ketiga berisi tentang penjelasan sistem yang dibangun. Bagian keempat mengenai analisi dan evaluasi sistem yang dibangun. Bagian ke lima berisi kesimpulan dan yang terakhir daftar pustaka.

2. Studi Terkait

2.1 Polling Menggunakan Internet

Pada tahun 2015 Hays, Liu, dan Kapteyn di dalam jurnalnya menyatakan bahwa penggunaan internet di dalam mengumpulkan data survei lebih murah dan membutuhkan waktu sedikit dibandingkan dengan metode tradisional. Penelitian tentang survei telah memasuki era baru, dengan sedikit penekanan terhadap wawancara dan meningkatkan penggunaan teknologi baru untuk mengupulkan datanya [5]. Namun, masih banyak hal yang perlu dipelajari mengenai keuntungan dan kerugian dalam pengunaan internet utnuk pengumpulan data survei, seperti dalam penggunaan web. Dan di masa depan terdapat peluang untuk perangkat seluler dan *platform* media sosial untuk melakukan hal yang sama.

Peningkatan kepemilikan dan penggunaan perangkat keras di kalangan mahasiswa, menciptakan peluang bagi fakultas untuk mengembangkan lingkungan belajar yang menarik. Dengan banyaknya Lembaga Pendidikan yang menawarkan akses Wi-Fi di seluruh kampus, sehingga mahasiswa memiliki kemampuan untuk menggunakan perangkat seluler (ponsel, tablet, atau laptop) mereka untuk terlibat pada pembelajaran, khususnya tentang konsep kepemimpinan. Noel, Stover, McNutt pada tahun 2015 mengusulkan *polling* berbasis *mobile* sebagai *Audience Response System* (ARS) [6]. Dan ternyata hasil dari pengalaman *polling* berbasis *mobile* menunjukan bahwa mahasiswa menjadi sangat terlibat dalam tiga level (perilaku, emosional, dan kognitif). Selain itu, tanggapan dari survei menyarankan *polling* berbasis *mobile* layak untuk dipakai di luar kelas[6].

Dari beberapa studi di atas, penulis juga di dalam penelitian ini menggunakan data *polling* yang didapatkan dari hasil *polling* menggunakan internet. Dimana *polling* dilakukan di sebuah web yang mengharuskan pemilih untuk login dengan *twitter* terlebih dahulu. Dengan memanfaatkan *twitter* API untuk mendapatkan data diri dari pemilih yang melakukan *polling*.

2.2 *Big Data*

Big data secara sederhana merupakan data yang memerlukan kapasitas pemprosesan melebihi pemprosesan pada sistem basis data konvensional. Secara umum, data yang masuk dalam kategori big data adalah data dengan volume melebihi satu tera-byte. Karakteristik yang dimiliki big data antara lain: *volume* (ukuran), *velocity* (kecepatan), *variety* (ragam), dan *veracity* (ketidakpastian data)[7].

2.3 *Apache Hadoop*

Pada tahun 2017, Merla dan Liang di dalam makalahnya telah melakukan penelitian mengenai pemanfaatan *Hadoop* di dalam pengolahan data dari media sosial *youtube*. Sistem yang dibangun dimulai dari ekstraksi data dari API *youtube*, menyimpan data ke *Hadoop Distributed File System* (HDFS), *Mapper* dan *Reducer* sampai hasil visualisasi dengan mengunakan *pie chart* menganai *trending* video berdasarkan kategori. Dari hasil pengolahan data youtube yang dilakukan dengan menggunakan *Hadoop* didapatkan sebuah kesimpulan berupa informasi trending video youtube bedasarkan kategori [8].

Hadoop merupakan tools yang paling popular dalam *Big Data*, yang banyak digunakan di jaringan sosial seperti, google [9]. Dengan *Hadoop* memungkinkan pemrosesan data berukuran besar secara terdistribusi dengan melibatkan berkluster-kluster komputer [4]. Hadoop memiliki beberapa kelebihan seperti efisensi dalam waktu eksekusi, meminimalkan biaya komputasi, dan meningkatkan kinerja [10]. Hadoop adalah *open source framework* yang digunakan untuk memproses data yang besar yang dilakukan secara parallel yang memeliki beberapa modul, yaitu diantaranya:

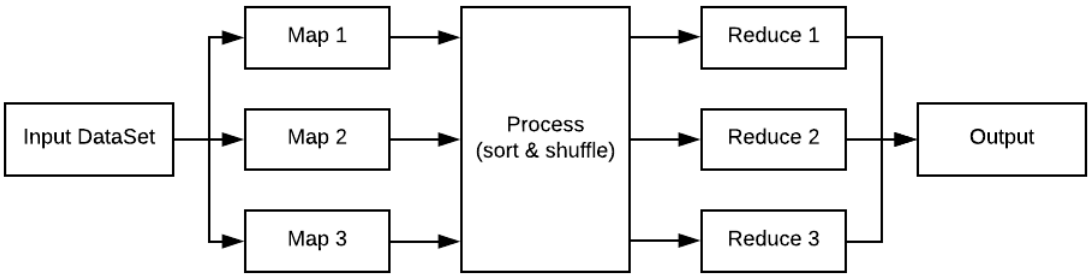
1. *Hadoop Distributed File System* (HDFS)
   HDFS adalah sistem penyimpanan terdistribusi yang dapat menyimpan filse secara terdistribusi pada HDFS node.
2. *Hadoop Common*
   Hadoop Common merupakan *libraries* dan *tools* yang dibutuhkan oleh modul *Hadoop* lainnya.
3. *Hadoop* YARN
   Hadoop YARN digunakan untuk mengatur *resource* yang akan digunakan.
4. *MapReduce*
   MapReduce adalah sebuah model program untuk teknik pengolahan data berdasarkan komputasi terdistribusi[3].

2.4 MapReduce

Model pemograman *MapReduce* digunakan unrtuk memproses secara efisien kumpulan data yang berukuran besars secara parallel. *MapReduce* terdiri dari tiga tahap yaitu, tahap map, shuffle, dan terakhir tahap *reduce*.

1. Tahap Map, memproses data inputan yang berupa file yang tersimpan di HDFS, file tersebut kemudian diubah menjadi tupel yaitu pasangan antara *key* dan *value*-nya.
2. Tahap Reduce, memproses data inputan dari hasil proses map, yang kemudian hasil data set barunya disimpan di HDFS kembali.

Untuk lebih jelasnya di dalam memahami tahapan dari proses *MapReduce* dapat dilihat pada Gambar 3.



Gambar 1 Proses Pemograman MapReduce

2.5 YARN (Resource Manager)

Untuk setiap *cluster* yang sangat besar dengan sekitar 4000 *nodes* atau lebih, sistem *MapReduce* yang ada sebelumnya memiliki problem dalam hal skalabilitas, jadi pada tahun 2010 sebuah grup di Yahoo! memulai untuk melakukan desain *MapReduce* generasi selanjurnya. Hasilnya adalah YARN, yang disingkat dari *Yet Another Resource Negotiator* [11].

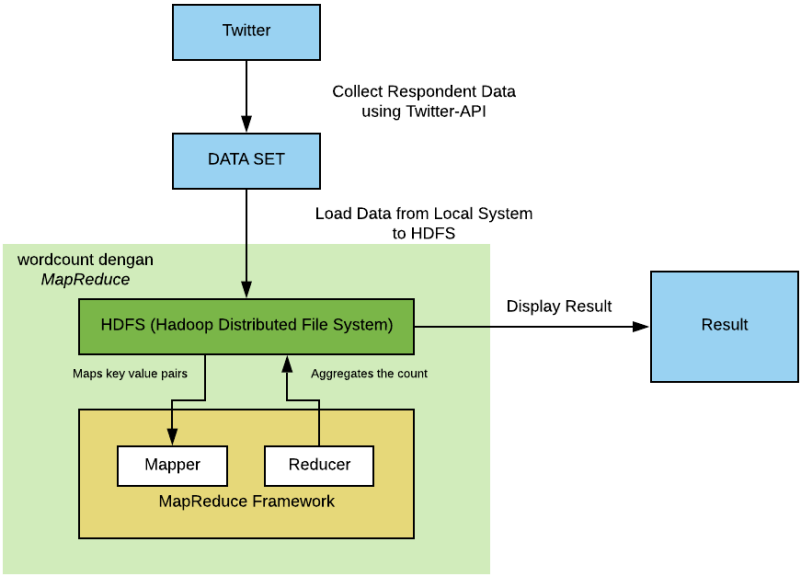
YARN menanggulangi kekurangan skalabilitas dari *MapReduce* klasik dengan membagi tanggung jawab dari *job tracker* kedalam beberapa bagian. *Jobtracker* yang bertanggung jawab terhadap *job scheduling* (mencocokkan *task* dengan *task trackers*) dan memonitor kemajuan dari *task* (tetap melacak tugas dan melakukan *restart* pada tugas yang lambat atau gagal, serta melakukan pembukuan seperti mencatat total tugas sejauh ini)[12].

2.6 Amazon EC2

Amazon Elastic Compute Cloud (Amazon EC2) adalah layanan web yang memberikan kapasitas komputasi yang aman dan berukuran fleksibel di cloud. Amazon EC2 dirancang untuk membuat komputasi cloud berskala web lebih mudah bagi pengembang. Antarmuka layanan *web* sederhana Amazon EC2 memungkinkan untuk mendapatkan dan mengonfigurasi kapasitas dengan friksi minimal. Amazon EC2 memberikan kendali penuh sumber daya komputasi dan memungkinkan bekerja di lingkungan komputasi Amazon yang telah terbukti [13].

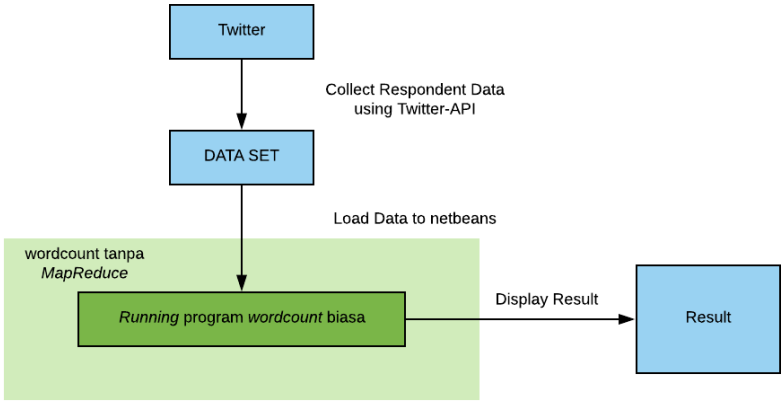
3. Perancangan dan Implementasi Sistem

Pada penelitian ini perancangan sistem yang akan dibangun didahului dengan proses instalasi dan konfigurasi *Hadoop-2.7.1* di sistem operasi Linux Ubuntu 14.04.6 LTS secara *MultiNode Cluster* di Amazon EC2. Setalah proses instalasi dan konfigurasi selesai, input data berupaa file .teks yang diinput dari data lokal ke HDFS. Data tersebut didapat dari hasil *polling* melalui aplikasi *web* happypoll.id, dimana voters harus *login* terlebih dahulu melalui media sosial *twitter*. Data yang telah diinput ke HDFS akan diproses oleh program *MapReduce* (*wordcount*). Adapun di dalam proses pengujiannya dilakukan beberapa kali percobaan dengan membandingkan program wordcount dengan *MaprReduce* pada *Hadoop* dan program *wordcount* yang biasa (tanpa *MapReduce* pada *Hadoop*). Untuk lebih jelasnya, proses data dengan menggunakan *MapReduce* pada *Hadoop* dapat dilihat pada Gambar 2.



Gambar 2 Proses Data dengan *MapReduce*

Dilakukan juga pengujian dengan program *wordcount* tanpa *MapReduce* pada *Hadoop*. Yaitu dengan pemograman java biasa proses nya dapat dilihat pada Gambar 3.



Gambar 3 Proses Data tanpa MapReduce

Setelah itu, dilakukan juga pengujian terhadap *Hadoop* yang melakukan proses data dengan ukuran *block default* yaitu 128 MB dan dengan ukuran *block* yang diperbesar menjadi 512 MB. Hal tersebut dilakukan untuk mengetahui pengaruh *block* pada HDFS terhadap kecepatan proses data.

3.1 Data yang Digunakan untuk Diproses

Data yang digunakan pada Tugas Akhir ini adalah, data yang berasal dari aplikasi *website* HappyPoll.id, dimana data tersebut merupakan data hasil *polling* yang dilakukan, dengan persoalan tentang Bapak Basuki Tjahaja Purnama (Ahok) yang ditunjuk Sebagai Komisaris Utama di Pertamina. Pada polling tersebut, responden dihadapkan dengan pilihan setuju atau tidak setuju dengan Bapak Basuki Tjahja Purnama yang terpilih sebagai Komisaris Utama di Pertamina. Data dari hasil jajak pendapat tersebut di-*export* ke bentuk *text file* (.text) untuk selanjutnya di-*load* ke HDFS dan dilakukan proses pengolahan dengan menggunakan pemograman *MapReduce* pada *framework Hadoop* dan program *wordcount* biasa dengan pemograman *java*. Penulis pada Tugas Akhir ini menggandakan data yang diperoleh ke dalam beberapa ukuran, yaitu data1.text dengan ukuran 55 MB, data2.text dengan ukuran 107,3 MB, data3.text dengan ukuran 214,7 MB, data4.text dengan ukuran 536,8 MB dan

data5.text dengan ukuran 1,07 GB. Hal ini dilakukan untuk mengetahui, kecepatan proses data pada program *wordcount* dengan *MapReduce Hadoop* dan *wordcount* tanpa *MapReduce*.

3.2 Pengolahan Data dengan *MapReduce*

Dari proses pengumpulan data, dilanjutkan dengan data di-*load* ke HDFS untuk selanjutnya diproses dengan komputasi secara terdistribusi oleh *MapReduce* pada *framework Hadoop*, dengan konfigurasi *multi node cluster*. Adapun untuk menjalankan perintah daripada proses *MapReduce* terdapat beberapa langkah untuk dilakukan, yaitu diantaranya:

1. Lakukan format pada *filesystem*:

```
$ bin/hdfs namenode -format
```

2. Jalankan *namenode* dan *datanode*:

```
$ sbin/start-dfs.sh
```

3. Buat direktori di HDFS untuk proses eksekusi *MapReduce jobs*:

```
$ bin/hdfs dfs -mkdir /user

$ bin/hdfs dfs -mkdir /user/<username>
```

4. *Copy file* dari lokal sistem ke HDFS:

```
$ bin/hdfs dfs -mkdir input

$ bin/hdfs dfs -put etc/hadoop/*.xml input
```

5. Jalankan *MapReduce*:

```
$ bin/hadoop jar share/hadoop/mapreduce/hadoop-mapreduce-examples-3.0.3.jar
grep input output 'dfs[a-z.]+'
```

Adapaun pada proses *MapReduce* sendiri ada beberapa tahapan proses yaitu, diantaranya:

1. Tahap Map, memproses data inputan yang berupa *file text* yang tersimpan di HDFS, file tersebut kemudian diubah menjadi tupel yaitu pasangan antara *key* dan *value*-nya.
2. Tahap Reduce, memproses data inputan dari hasil proses map, yang kemudian hasil data set barunya disimpan di HDFS kembali untuk proses selanjutnya.

3.3 Pengujian Apliakasi *Wordcount*

Dilakukan 2 pengujian aplikasi *wordcount*, yaitu program *wordcount* dengan *MapRedue* dan tanpa *MapReduce*. Hal ini dilakukan untuk mengetahui mana yang lebih efektif untuk memproses data *polling* yang didapat dari dilakukannya *polling* di website happypoll.id yang berbasis media sosial.

Pengujian tersebut dilakukan dengan melihat kecepetan proses data pada setiap program, di setiap data yang diproses. Terdapat 5 data yang diproses untuk mengetahui kecepetan masing-masing program.

3.4 Komponen Perangkat Lunak yang Digunakan untuk Membangun Sistem

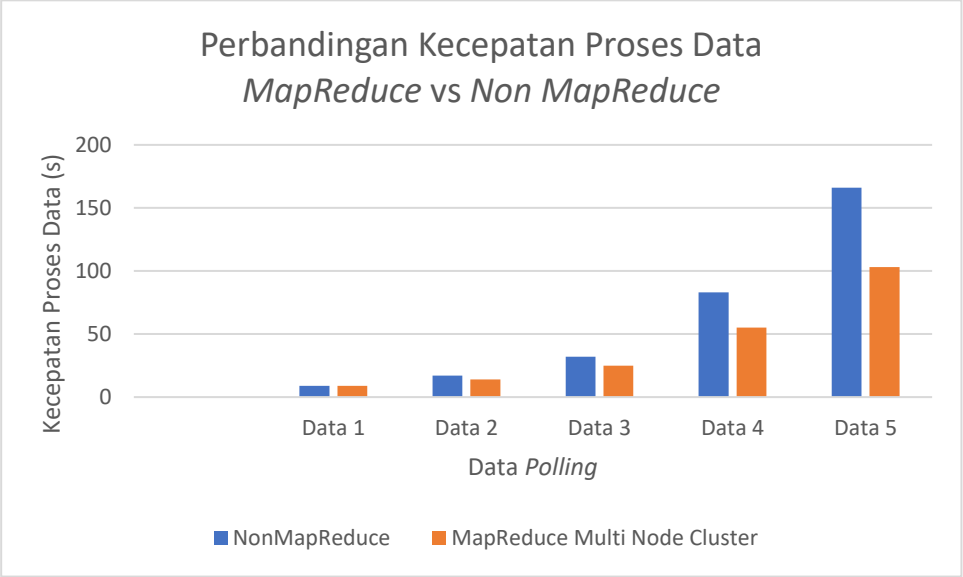
Adapaun perangkat lunak yang digunakan di dalam pengimplementasian sistem ini adalah sebagai berikut:

1. Sistem Operasi
   Sistem Operasi yang digunakan pada komputer adalah Linux Ubuntu 14.04.6 LTS
2. Java
   Java digunakan untuk menunjang penggunaan *Hadoop Distributed File System*. Pada *Hadoop cluster* diperlukan Java *Run Time Environment* (JRE) dan *Java Development Kit* (JDK) dengan versi Java jdk-1.8.0.
3. Hadoop
   Hadoop yang digunakan adalah Hadoop-2.7.1 dengan *setting multi node cluster*.
4. Open SSH
   Hadoop berjalan di atas server SSH.
5. Aplikasi Web
   Untuk mendapatkan data jajak pendapat, dimana *voters* yang melakukan jajak pendapat diharuskan untuk *login* terlebih dahulu dengan *twitter*.
6. Google Chrome
   Digunakan untuk mengakses HDFS dan *recourcemanager*.
7. Amazon EC2
   Digunakan untuk instalasi *Hadoop multi node cluster*, dan untuk memproses data yang besar yang tidak bisa dilakukan oleh hanya komputer penulis.

4. Evaluasi

4.1 Perbandingan Kecepatan Proses Data dengan *MapReduce dan Tanpa MapReduce*

Pada Tugas Akhir ini juga dilakukan beberapa proses data, dengan mengujikan 5 data dengan ukuran yang berbeda-beda. Terhadap 2 program, yaitu program *wordcount* dengan *MapReduce* dan program *wordcount* tanpa *MapReduce*. Hal tersebut untuk mengetahui seberapa cepat kedua program tersebut di dalam memproses data *polling* berbasis media sosial yang didapat oleh penulis. Adapun hasil yang didapat dapat dilihat dari gambar 4.



Gambar 4 Perbandingan Kecepatan Proses Data *MapReduce* vs *Non MapReduce*

Dari grafik diatas, untuk data 1 program *wordcount* dengan *MapReduce* membutuhkan waktu 9 detik sedangkan tanpa *MapReduce* juga sama membutuhkan waktu 1 detik untuk proses data. Untuk data 2 pada program *wordcount* dengan *MapReduce* membutuhkan waktu 14 detik sedangkan tanpa *MapReduce* membutuhkan 17 detik. Untuk data 3 pada program *wordcount* dengan *MapReduce* membutuhkan waktu 25 detik sedangkan tanpa *MapReduce* membutuhkan 32 detik. Untuk data 4 pada program *wordcount* dengan *MapReduce* membutuhkan waktu 55 detik sedangkan tanpa *MapReduce* membutuhkan 83 detik. Dan yang terakhir untuk data 5 pada program *wordcount* dengan *MapReduce* hanya membutuhkan waktu 103 detik sedangkan tanpa *MapReduce* membutuhkan 166 detik. Untuk lebih jelasnya dapat dilihat pada tabel berikut.

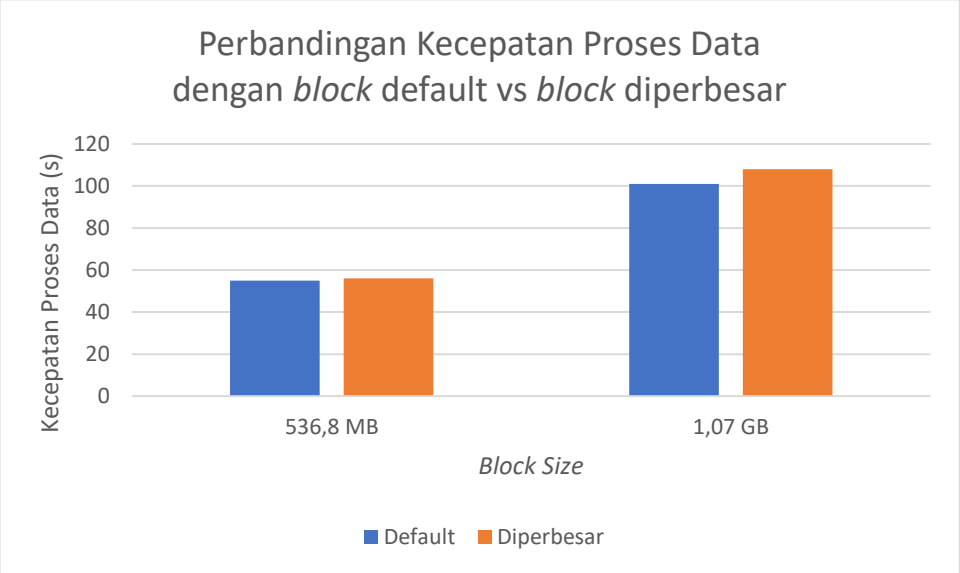
Tabel 1 Perbandingan Kecepatan Proses Data

Data	*NonMapReduce*	*MapReduce Multi Node Cluster*
Data 1 (55 MB)	9 detik	9 detik
Data 2 (107,3 MB)	17 detik	14 detik
Data 3 (214,7 MB)	32 detik	25 detik
Data 4 (536,8 MB)	83 detik	55 detik
Data 5 (1,07 GB)	166 detik	103 detik

Dari ke-5 data yang diujikan, diketahui bahwa program *wordount* dengan *MapReduce Hadoop* efekti digunakan untuk proses data yang besar. Meskipun pada data 1 yang ukuranya lebih kecil kedua program membuthkan waktu yang sama, akan tetapi setelah dilakukan beberapa kali uji dengan data yang lebih besar program *wordcount* dengan *MapReduce* unggul. Semakin besar data yang diproses maka program *wordcount* dengan *MapReduce* semakin unggul dengan memperlebar jarak perbandingan proses data berdasarkan waktu.

4.2 Perbandingan Kecepatan Proses Data Ukuran *Block* 128MB dengan ukuran *Block* 512MB

Setelah dilakukan pengujian terhadap *Hadoop* dengan ukuran *block* yang diperbesar diketahui bahwa, *block* yang diperbesar tidak lebih cepat dibandingkan dengan *block* yang ukuran *default* (128MB). Untuk lebih jelasnya dapat dilihat pada gambar 5.



Gambar 5 Perbandingan Kecepatan Proses Data dengan Ukuran Block yang Diperbesar

Dari grafik tersebut, *block* dengan ukuran *default* untuk memproses data 536,8 MB membutuhkan waktu 55 detik, sedangkan dengan *block* yang diperbesar membutuhkan waktu 56 detik. Dan untuk data yang berukuran 1,07 GB *block* dengan ukuran *default* membutuhkan waktu 101 detik dan untuk *block* yang diperbesar membutuhkan waktu 108 detik.

4.3 Hasil Analisis dari Pengujian

Dari pengujian yang dilakukan, pemrosesan data *polling* dengan memanfaatkan *MapReduce* pada *Framework Hadoop* untuk data yang didapat penulis optimal untuk digunakan. Hal tersebut diketahui dari pemrosesan data *polling* yang telah dilakukan, dari beberapa data yang diujikan untuk proses data dengan program *wordcount* dengan *MapReduce* pada *Hadoop* membutuhkan waktu yang relatif singkat untuk data yang berukuran besar. Hal tersebut dikarenakan karakterisitik dari *Hadoop* sendiri yang diperuntukan untuk memproses data secara distribusi, sehingga proses data akan lebih cepat dengan menggunakan *Hadoop*.

Dan dari hasil perbandingan yang dilakukan dengan membandingakan program *wordcount* dengan dan tanpa *MapReduce*. Menghasilkan bahwa program *wordcount* untuk proses data *polling* unggul dengan program dengan *MapReduce Hadoop*. Dari pengujian, semakin besar data untuk metode tanpa *Hadoop* performansinya semakin menurun, dengan proses data yang cukup lama. Sedangkan untuk program *wordcount* dengan *MapReduce Hadoop* meskipun data yang diproses semakin besar performansi di dalam memproses data stabil. Hal itu berdasarkan dengan semakin besar data yang diperoleh semakin besar jarak waktu yang dibutuhkan antara program *wordcount* dengan *Hadoop* dan *wordcount* tanpa *Hadoop*. Dengan menggunakan *Hadoop* dapat memproses data 1,3 kali lebih cepat dibandingkan tanpa *Hadoop*.

Untuk pengujian terhadap *block size* setalah dilakukan pengujian terhadap data yang berukuran 53,6 MB dan 1,07 GB menghasilkan ukuran *block* yang *default* lebih unggul sedikit dibandingkan dengan ukuran *block* yang diperbesar.

5. Kesimpulan dan Saran

5.1 Kesimpulan

Berdasarkan analisis dari hasil uji yang dilakukan pada sistem yang dibangun pada Tugas Akhir ini terdapat beberapa kesimpulan, yaitu diantaranya:

1. *MapReduce* pada Hadoop otimal digunakan untuk memproses data *polling* karena dalam proses data membutuhkan waktu yang relatif sedikit.
2. *Wordcount* yang merupakan program sederhana dari *MapReduce* cocok digunakan untuk kasus data *polling*.
3. Dari 5 data yang diproses dengan ukuran yang berbeda-beda, *wordcount* dengan *MapReduce Hadoop* unggul dibandingkan dengan program *wordcount* biasa. Dengan rata-rata dari pengujian 5 data, *MapReduce Hadoop* dapat 1,3 kali lebih cepat dibandingkan dengan tanpa *MapReduce Hadoop*.
4. Ukuran *block* HDFS yang diperbesar mempengaruhi kecepatan proses data, namun hasilnya lebih lama dibandingkan dengan ukuran *block* yang *default*.

5.2 Saran

Adapun terdapat beberapa saran untuk pengembangan daripada Tugas Akhir ini, yaitu dengan memanfaatkan *streaming* data pada *Hadoop* untuk mendapatkan data yang lebih besar. Dan bandingkan *Hadoop* dengan metode lainnya seperti *apache spark*, *mongdb* dll.

Daftar Pustaka

[1] H. L. Li, V. T. Y. Ng, and S. C. K. Shiu, "Predicting short interval tracking polls with online social media," *Proc. 2013 IEEE 17th Int. Conf. Comput. Support. Coop. Work Des. CSCWD 2013*, no. Idc, pp. 587–592, 2013.

[2] P. M. Bante and K. Rajeswari, "Big Data Analytics Using Hadoop Map Reduce Framework and Data Migration Process," *2017 Int. Conf. Comput. Commun. Control Autom. ICCUBEA 2017*, pp. 1–5, 2018.

[3] T. Advancements, G. Jagdev, B. Singh, and M. Mann, "Subcontinent," *Int. J. Sci. Tech. Adv.*, vol. 1, no. 3, 2015.

[4] C. Verma and R. Pandey, "Big Data representation for grade analysis through Hadoop framework," *Proc. 2016 6th Int. Conf. - Cloud Syst. Big Data Eng. Conflu. 2016*, pp. 312–315, 2016.

[5] R. D. Hays, H. Liu, and A. Kapteyn, "Use of Internet panels to conduct surveys," *Behav. Res. Methods*, vol. 47, no. 3, pp. 685–690, 2015.

[6] D. Noel, S. Stover, and M. McNutt, "Student perceptions of engagement using mobile-based polling as an audience response system : Implications for leadership studies," *J. Leadersh. Educ.*, no. Summer, pp. 53–70, 2015.

[7] G. Kapil, A. Agrawal, and R. A. Khan, "A study of big data characteristics," *Proc. Int. Conf. Commun. Electron. Syst. ICCES 2016*, 2016.

[8] P. R. Merla and Y. Liang, "Data analysis using hadoop MapReduce environment," *Proc. - 2017 IEEE Int. Conf. Big Data, Big Data 2017*, pp. 4783–4785, 2017.

[9] K. Rattanaopas and S. Kaewkeeree, "Improving Hadoop MapReduce performance with data compression: A study using wordcount job," *ECTI-CON 2017 - 2017 14th Int. Conf. Electr. Eng. Comput. Telecommun. Inf. Technol.*, pp. 564–567, 2017.

[10] S. R. Suthar, V. K. Dabhi, and H. B. Prajapati, "Machine learning techniques in Hadoop environment: A survey," *2017 Innov. Power Adv. Comput. Technol. i-PACT 2017*, vol. 2017-Janua, pp. 1–8, 2017.

[11] K. Basuki, H. N. Palit, and L. P. Dewi, "Implementasi Hadoop : Studi Kasus Pengolahan Data Peminjaman Perpustakaan Universitas Kristen Petra," 2015.

[12] T. C. Bressoud and Q. Tang, "Results of a model for Hadoop YARN MapReduce tasks," *Proc. - IEEE Int. Conf. Clust. Comput. ICCC*, pp. 443–446, 2016.

[13] N. Ekwe-Ekwe and A. Barker, "Location, location, location: Exploring amazon EC2 spot instance pricing across geographical regions," *Proc. - 18th IEEE/ACM Int. Symp. Clust. Cloud Grid Comput. CCGRID 2018*, pp. 370–373, 2018.