

SKRIPSI

**STUDI DAN IMPLEMENTASI APACHE SPARK MLLIB
UNTUK ANALISIS BIG DATA**



Kresna Dwi Cahyo

NPM: 2014730048

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS
UNIVERSITAS KATOLIK PARAHYANGAN
2018**

UNDERGRADUATE THESIS

**STUDY AND IMPLEMENTATION OF APACHE SPARK
MLLIB FOR BIG DATA ANALYSIS**



Kresna Dwi Cahyo

NPM: 2014730048

**DEPARTMENT OF INFORMATICS
FACULTY OF INFORMATION TECHNOLOGY AND SCIENCES
PARAHYANGAN CATHOLIC UNIVERSITY
2018**

LEMBAR PENGESAHAN



STUDI DAN IMPLEMENTASI APACHE SPARK MLLIB UNTUK ANALISIS BIG DATA

Kresna Dwi Cahyo

NPM: 2014730048

Bandung, 28 Mei 2018

Menyetujui,

Pembimbing Utama

Dr. Veronica Sri Moertini

Pembimbing Pendamping

Gede Karya, M.T., CISA

Ketua Tim Penguji

Vania Natali, M.T.

Anggota Tim Penguji

Rosa De Lima, M.Kom.

Mengetahui,

Ketua Program Studi

Mariskha Tri Adithia, P.D.Eng



PERNYATAAN

Dengan ini saya yang bertandatangan di bawah ini menyatakan bahwa skripsi dengan judul:

STUDI DAN IMPLEMENTASI APACHE SPARK MLLIB UNTUK ANALISIS BIG DATA

adalah benar-benar karya saya sendiri, dan saya tidak melakukan penjiplakan atau pengutipan dengan cara-cara yang tidak sesuai dengan etika keilmuan yang berlaku dalam masyarakat keilmuan.

Atas pernyataan ini, saya siap menanggung segala risiko dan sanksi yang dijatuhkan kepada saya, apabila di kemudian hari ditemukan adanya pelanggaran terhadap etika keilmuan dalam karya saya, atau jika ada tuntutan formal atau non-formal dari pihak lain berkaitan dengan keaslian karya saya ini.

Dinyatakan di Bandung,
Tanggal 28 Mei 2018



Kresna Dwi Cahyo
NPM: 2014730048

ABSTRAK

Data dalam waktu cepat dapat terkumpul dapat menjadi Big Data. Big data dapat dianalisis untuk mendapatkan informasi atau pengetahuan yang berharga. Namun, dibutuhkan komputer dengan kekuatan komputasi yang sangat tinggi untuk menganalisis data dengan ukuran yang sangat besar. Sistem terdistribusi adalah solusi dari masalah tersebut. Salah satu jenis sistem terdistribusi adalah *Distributed Computing System*. *Distributed Computing System* merupakan sistem terdistribusi yang digunakan untuk komputasi dengan kebutuhan kinerja yang tinggi. Hadoop adalah salah satu *framework* yang banyak dipakai saat ini. Hadoop merupakan platform yang dapat menyediakan penyimpanan dan kemampuan komputasi terdistribusi. Seiring berjalannya waktu, *Framework* lainnya juga mulai bermunculan. Salah satunya adalah Apache Spark.

Apache Spark adalah sebuah *framework cluster computing* yang dapat dimanfaatkan untuk mengolah Big Data dengan cepat. Apache Spark dapat berjalan diatas infrastruktur Hadoop untuk meningkatkan fungsionalitas. MLlib merupakan *library* yang mengimplementasikan teknik analisis *machine learning*. Teknik *machine learning* yang dimiliki MLlib meliputi *statistic*, *classification*, *regression*, *collaborative filtering*, *clustering*, *dimensionality reduction*, dan *feature extraction*. Fungsi-fungsi pada MLlib dapat dipanggil melalui API yang disediakan Spark.

Perangkat lunak demo berhasil dikembangkan pada skripsi ini. Perangkat Lunak ini memiliki tujuan untuk dapat mengolah data dengan menggunakan beberapa fungsi dari *library* MLlib. Perangkat lunak demo berjalan pada lingkungan *cluster* hadoop. Skripsi ini menjelaskan konsep dan cara kerja Spark serta menunjukkan cara pemanggilan fungsi-fungsi MLlib. Selain itu, modifikasi dilakukan pada fungsi K-means MLlib agar dapat menghasilkan suatu pola bagi pengguna. Pengujian fungsional dan uji performa dilakukan untuk menguji fungsi-fungsi *library* MLlib. Uji performa dilakukan dengan variasi ukuran data masukkan yaitu 1gb, 3gb, dan 5gb. Berdasarkan hasil pengujian yang sudah dilakukan, dapat disimpulkan bahwa kinerja dari fungsi-fungsi MLlib sangat baik untuk komputasi pada ukuran data yang besar.

Kata-kata kunci: Sistem Terdistribusi, Apache Spark, MLlib, Machine Learning, Hadoop, HDFS, Clustering, K-means, Scala

ABSTRACT

Data can quickly gathered and will become Big Data in a short time. Big Data can be analyzed to gain valuable information or a knowledge. However, it requires a computer with high power of computing to analyzed large data. Distributed system is the solution. One type of distributed system is the Distributed Computing System. Distributed Computing System is a distributed system that used for high performance computing. Hadoop is one of the framework that commonly used. Hadoop is a platform that provide storage system and computational capability. As time goes by, another framework is developed. One of them is Apache Spark.

Apache Spark is a cluster computing platform that can be utilized to processing Big Data faster. Apache Spark can run on hadoop environment to gain functionality. MLlib is a Spark library which implements machine learning analysis technique. Such as statistic, classification, regression, collaborative filtering, clustering, dimensionality reduction, and feature extraction. This MLlib functions can be called through the API provided by Spark.

The demo software was successfully developed in this essay. It has a purpose to be able to process data by calling function from MLlib library. The demo software runs on hadoop cluster environment. This essay explain spark concepts, how spark works, and shows how to call MLlib function. In addition, modifications are made to the K-means MLlib library in order to generate patterns for users. Functional tests and performance tests are performed to test functions of MLlib library. Performance test has been done with varying input data size such as 1gb, 3gb, and 5gb. Based on the tests results that have been done, it can be concluded that the performance of MLlib functions is good for computing on large scale data size.

Keywords: Distributed System, Apache Spark, MLlib, Machine Learning, Hadoop, HDFS, Clustering, K-means, Scala

Dipersembahkan kepada keluarga tercinta

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Esa atas segala rahmat dan berkat-Nya penulis dapat menyelesaikan skripsi yang berjudul "Studi dan Implementasi Apache Spark MLlib untuk Analisis Big Data". Dalam proses penyusunan skripsi, penulis banyak mendapat kesempatan untuk mempelajari hal-hal baru, serta mendapatkan banyak bantuan dan dukungan baik secara langsung maupun tidak langsung dari berbagai pihak. Oleh karena itu, pada kesempatan ini penulis ingin mengucapkan terima kasih kepada seluruh pihak yang telah berperan dalam penyusunan skripsi ini. Secara khusus penulis ingin menyampaikan terima kasih kepada:

1. Tuhan Yesus Kristus yang selalu menyertai penulis baik suka maupun duka.
2. Orang tua penulis yaitu R. Sudharma Basuki dan Henrina B., serta kakak dan adik penulis, dan keluarga besar penulis keluarga besar Soekartomo dan keluarga besar Kadang yang selalu mendoakan dan mendukung penulis selama penyusunan skripsi.
3. Ibu Dr. Veronica Sri Moertini dan Bapak Gede Karya, M.T., CISA selaku dosen pembimbing yang telah memberi arahan dan masukan selama penyusunan skripsi.
4. Ibu Vania Natali, M.T. dan Ibu Rosa De Lima, M.Kom. selaku dosen penguji yang telah memberikan kritik dan saran untuk skripsi.
5. Rekan-rekan di Informatika Unpar yaitu Ivan T.W, Prayogo C., Kevin P, Kevin Jonathan, Kelvin, Daniel F., Stillmen S., Melinda N., rekan-rekan big data, rekan-rekan kosan, rekan-rekan *touring*, rekan-rekan admin lab FTIS dan lain-lain; atas dukungan yang diberikan kepada penulis.
6. Pihak-pihak lain yang belum disebutkan, yang telah memberikan bantuan dalam penyusunan skripsi.

Akhir kata, penulis menyadari bahwa skripsi ini tidak lepas dari kekurangan. Namun penulis berharap skripsi ini dapat memberikan kontribusi baik untuk penelitian atau pembelajaran selanjutnya.

Bandung, Mei 2018

Penulis

DAFTAR ISI

KATA PENGANTAR	xv
DAFTAR ISI	xvii
DAFTAR GAMBAR	xix
DAFTAR TABEL	xxi
1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan	2
1.4 Batasan Masalah	2
1.5 Metodologi	3
1.6 Sistematika Pembahasan	3
2 LANDASAN TEORI	5
2.1 Sistem Terdistribusi	5
2.1.1 Pengertian Sistem Terdistribusi	5
2.1.2 Jenis Sistem Terdistribusi	6
2.2 Big Data	6
2.3 Hadoop	6
2.3.1 MapReduce	8
2.3.2 Hadoop Distributed File System (HDFS)	9
2.3.3 Blok HDFS	12
2.4 Apache Spark	13
2.4.1 Komponen Apache Spark	15
2.4.2 Arsitektur Apache Spark	17
2.4.3 Resilient Distributed Datasets (RDD)	19
2.4.4 MLib	21
2.5 Scala	29
2.5.1 Tipe Data	30
2.5.2 Variable	30
2.5.3 Method dan Fungsi	31
2.5.4 Iterasi dan Pencabangan	32
2.5.5 Objek	33
2.5.6 Method Main	33
3 STUDI DAN EKSPLORASI APACHE SPARK MLlib	35
3.1 Instalasi Apache Spark	35
3.2 Eksplorasi menggunakan Spark Shell	36
3.3 Pemanggilan Fungsi-Fungsi pada library MLlib	37
3.3.1 Naive Bayes	40

3.3.2	K-Means	42
3.3.3	Summary Statistics	44
3.3.4	Principal Component Analysis (PCA)	45
3.3.5	Alternating Least Squares (ALS)	47
3.3.6	Term Frequency - Inverse Document Frequency (TF-IDF)	49
4	PENGEMBANGAN K-MEANS PADA LINGKUNGAN SPARK	51
4.1	Kebutuhan Pengembangan	51
4.2	Analisis K-Means MLib	52
4.3	Modifikasi K-Means MLib	57
5	PERANCANGAN, IMPLEMENTASI, DAN PENGUJIAN PERANGKAT LUNAK DEMO	63
5.1	Diagram Use Case dan Skenario	63
5.2	Diagram Kelas dan Pemanggilan Library MLib	67
5.3	Rancangan User Interface	72
5.4	Implementasi Perangkat Lunak	78
5.4.1	Lingkungan Perangkat Keras	78
5.4.2	Lingkungan Perangkat Lunak	78
5.4.3	User Interface	79
5.4.4	Implementasi Pemanggilan Fungsi MLib	82
5.5	Pengujian Perangkat Lunak	88
5.6	Eksperimen untuk Uji Performa Fungsi-fungsi MLib	99
5.6.1	Lingkungan Perangkat Keras	99
5.6.2	Lingkungan Perangkat Lunak	100
5.6.3	Hasil Eksperimen	100
6	KESIMPULAN DAN SARAN	107
6.1	Kesimpulan	107
6.2	Saran	107
	DAFTAR REFERENSI	109
A	KODE PROGRAM IMPLEMENTASI RANCANGAN PERANGKAT LUNAK DEMO ML-LIB	111
B	HASIL EKSPERIMEN UJI PERFORMA	133

DAFTAR GAMBAR

2.1	Sistem terdistribusi yang diorganisir <i>middleware</i>	5
2.2	Arsitektur Hadoop	8
2.3	Arsitektur HDFS : Client berkomunikasi dengan NameNode(Master) dan DataNode(slave)	9
2.4	Arsitektur HDFS : Client berkomunikasi dengan NameNode(Master) dan DataNode(slave)	10
2.5	Diagram ekosistem data processing Spark	13
2.6	Tahap pemrosesan data pada Spark	14
2.7	Tahap pemrosesan data pada MapReduce Hadoop	14
2.8	Tiga cara untuk melakukan deployment pada Spark	15
2.9	Apache Spark stack	16
2.10	Arsitektur Apache Spark	17
2.11	RDD pada sebuah cluster	19
2.12	Proses yang umum terjadi pada RDD	20
2.13	Scatterplot contoh data dua dimensi	27
2.14	Bentuk umum <i>function</i> Scala	31
3.1	Tampilan Spark-Shell	37
3.2	Membuat <i>project</i> SBT pada IntelliJ	39
3.3	Mengatur konfigurasi project pada IntelliJ	40
3.4	Hasil Naive Bayes dikeluarkan pada console	41
3.5	Hasil Naive Bayes dikeluarkan pada console	42
3.6	Hasil output program eksperimen KMeans	43
3.7	Hasil output percobaan Summary Statistic	45
3.8	Hasil keluaran eksperimen PCA pada console	47
3.9	Hasil output prediksi	49
3.10	Data masukan untuk TF-IDF	49
3.11	Hasil output TF-IDF pada console	50
4.1	Diagram flow iterasi K-Means pada MLlib	55
5.1	Diagram <i>use case</i> perangkat lunak demo	64
5.2	Diagram kelas perangkat lunak demo	68
5.3	Tampilan pengolahan data menggunakan algoritma Naive Bayes	73
5.4	Tampilan pengolahan data menggunakan algoritma K-Means	74
5.5	Tampilan pengolahan data menggunakan metode Statistik	75
5.6	Tampilan pengolahan data menggunakan algoritma PCA	76
5.7	Tampilan pengolahan data menggunakan algoritma ALS	77
5.8	Tampilan pengolahan data menggunakan algoritma TF-IDF	78
5.9	Tampilan pengolahan data menggunakan algoritma Naive Bayes	79
5.10	Tampilan pengolahan data menggunakan algoritma K-Means	80
5.11	Tampilan pengolahan data menggunakan metode Statistik	80
5.12	Tampilan pengolahan data menggunakan algoritma PCA	81

5.13	Tampilan pengolahan data menggunakan algoritma ALS	81
5.14	Tampilan pengolahan data menggunakan algoritma TF-IDF	82
5.15	Data masukkan untuk fitur naive bayes	91
5.16	Data masukkan untuk fitur k-means	92
5.17	Data masukkan untuk fitur Statistik dan PCA	92
5.18	Data masukkan untuk fitur ALS	93
5.19	Data masukkan untuk fitur TF-IDF	93
5.20	Hasil pengujian fungsional naive bayes model	94
5.21	Hasil pengujian fungsional prediksi dengan model naive bayes	94
5.22	Hasil pengujian fungsional K-Means	95
5.23	Hasil pengujian fungsional statistik	96
5.24	Hasil pengujian fungsional PCA	97
5.25	Hasil pengujian fungsional model ALS	98
5.26	Hasil pengujian fungsional prediksi dengan model ALS	98
5.27	Hasil pengujian fungsional TF-IDF	99
5.28	Arsitektur lingkungan cluster Hadoop untuk eksperimen	100
5.29	Grafik perhitungan waktu eksekusi pembuatan model Naive Bayes	101
5.30	Grafik perhitungan waktu eksekusi prediksi data dengan model Naive Bayes	101
5.31	Grafik perhitungan waktu eksekusi eksperimen K-Means	102
5.32	Grafik perhitungan waktu eksekusi eksperimen Statistik	103
5.33	Grafik perhitungan waktu eksekusi eksperimen PCA	104
5.34	Grafik perhitungan waktu eksekusi pembuatan model Alternating Square Least	105
5.35	Grafik perhitungan waktu eksekusi prediksi data dengan model Alternating Square Least	105
5.36	Grafik perhitungan waktu eksekusi eksperimen TF-IDF	106

DAFTAR TABEL

2.1	Tipe data pada bahasa pemrograman Scala [1]	30
5.1	Tabel keterkaitan kelas pada perangkat lunak demo dan <i>library</i> MLib	69
5.2	Hasil Pengujian Fungsional Perangkat Lunak	89
5.3	Tabel 'pola' centroid pertama hasil eksekusi perangkat lunak	89
5.4	Tabel 'pola' centroid kedua hasil eksekusi perangkat lunak	90
5.5	Tabel 'pola' centroid ketiga hasil eksekusi perangkat lunak	90
5.6	Tabel 'pola' centroid pertama hasil eksekusi MATLAB	90
5.7	Tabel 'pola' centroid kedua hasil eksekusi MATLAB	90
5.8	Tabel 'pola' centroid ketiga hasil eksekusi MATLAB	90
B.1	Hasil Pengujian Eksekusi Fungsi MLib pada Perangkat Lunak	133
B.2	Hasil Pengujian Prediksi Data Baru Fungsi MLib pada Perangkat Lunak	133

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Seiring dalam pemanfaatan teknologi informasi yang makin meluas, data dengan cepat terkumpul, sehingga dalam waktu yang singkat dapat mencapai ratusan gigabyte. Sebagai contoh, beberapa data yang dapat terkumpul dengan cepat adalah data log akses ke suatu web, data teks(email, status facebook, dll), data waktu dan lokasi, data sensor, dan masih banyak lagi. Data tersebut dinamakan "Big Data". Big data menurut Gartner(2012) didefinisikan sebagai data yang memiliki ukuran(*volume*), kecepatan(*velocity*), dan/atau ragam (*variety*) yang ekstrim, yang menuntut pemrosesan informasi yang cepat dan inovatif untuk mendukung pengambilan keputusan dan otomatisasi proses. Terdapat tiga jenis big data yaitu *structured data*, *unstructured data*, *semi-structured data*.

Big data dapat dianalisis untuk mendapatkan informasi atau pengetahuan yang berharga. Beberapa teknik analisis big data di antaranya adalah peringkasan data yang berbasis Association Rule Learning, Classification tree analysis, Genetic algorithms, Machine learning, Regression analysis, Sentimental Analysis, dan Social network analysis.

Hadoop merupakan platform *open source* yang dapat menyediakan penyimpanan dan kemampuan komputasi terdistribusi. Dengan Hadoop, pengguna dapat menyimpan big data dan melakukan komputasi analisis terhadap big data. Komponen utama dari hadoop yaitu HDFS(Hadoop Distributed File System) dan MapReduce. HDFS merupakan suatu *filesystem* yang bertujuan untuk menyimpan data yang berukuran besar secara terdistribusi. MapReduce melakukan pengolahan data pada Hadoop.

Apache Spark adalah sebuah platform *cluster computing* yang dapat dimanfaatkan untuk mengolah big data dengan cepat. Spark merupakan pengembangan model MapReduce untuk melakukan komputasi agar lebih cepat dan efisien. Apache Spark dapat berjalan pada infrastruktur Hadoop salah satunya adalah HDFS. Apache Spark dibuat menggunakan bahasa pemrograman Scala dan berjalan pada Java Virtual Machine (JVM). Beberapa *library* di dalam Spark adalah Spark SQL, Spark Streaming, MLlib, dan GraphX.

MLlib merupakan library yang mengimplementasikan teknik analisis *machine learning* pada big data. Beberapa jenis algoritma *machine learning* yang disediakan di antaranya adalah *classification*, *regression*, *clustering*, *collaborative filtering*, dll. Semua algoritma tersebut didesain untuk dapat dioperasikan pada cluster hadoop. Pada algoritma clustering MLlib didapati bahwa K-Means perlu dimodifikasi agar dapat menghasilkan pola.

Pada penelitian ini dipelajari konsep Apache Spark dan library MLlib untuk melakukan pengolahan big data. Dalam mendukung hal tersebut, pada skripsi ini dibangun sebuah perangkat lunak demo untuk mengakses fungsi-fungsi pada MLlib agar MLlib lebih mudah diakses dan digunakan. Selain itu, algoritma K-Means MLlib dimodifikasi agar dapat menghasilkan pola yang bermanfaat

bagi pengguna.

1.2 Rumusan Masalah

Rumusan masalah pada penelitian ilmiah ini adalah :

1. Bagaimana cara kerja konsep Apache Spark dan Hadoop?
2. Bagaimana cara kerja fungsi-fungsi *library* MLib pada Apache Spark?
3. Bagaimana modifikasi dari algoritma K-Means paralel pada Spark MLib?
4. Bagaimana performa Apache Spark MLib dengan menggunakan *big data*?
5. Bagaimana pengembangan perangkat lunak demo MLib?

1.3 Tujuan

Berdasarkan rumusan masalah tersebut, tujuan dari penelitian ini adalah sebagai berikut:

1. Memahami konsep framework Apache Spark dan Hadoop
2. Memahami cara kerja *library* MLib untuk analisis *big data* dan melakukan uji coba pemanfaatan/pemanggilan fungsi-fungsi MLib
3. Merancang modifikasi yang dilakukan pada source code K-Means MLib
4. Menguji performa Apache Spark MLib dengan menggunakan big data
5. Mengembangkan perangkat lunak untuk memproses data yang memanggil fungsi-fungsi pada MLib untuk analisis data dan menampilkan hasil analisis data tersebut.

1.4 Batasan Masalah

Batasan masalah pada Skripsi ini adalah:

1. Studi literatur konsep Hadoop hanya dilakukan pada dasar Hadoop dan konsep *filesystem* Hadoop yaitu HDFS.
2. Studi literatur Apache Spark hanya dilakukan pada komponen Apache Spark yaitu *library* MLib.
3. Beberapa fungsi MLib yang dipelajari adalah fungsi Statistik, Naive Bayes, K-Means, Principal Component Analisis (PCA), Alternating Least Square (ALS), dan Term frequency-inverse document frequency (TF-IDF).
4. Studi literatur bahasa pemrograman scala hanya pada dasar-dasar pada bahasa pemrograman scala untuk memahami cara kerja Apache Spark.
5. Pengujian pada skripsi ini di batasi pada pengujian fungsional untuk menunjukkan berfungsinya pemanggilan fungsi MLib pada pengembangan perangkat lunak demo dan uji performa dengan variasi ukuran data masukkan sebesar 1gb, 3gb, dan 5gb.

1.5 Metodologi

Penyusunan Skripsi ini menggunakan metodologi sebagai berikut:

1. Melakukan studi literatur tentang konsep Apache Spark.
2. Melakukan studi literatur penggunaan Hadoop terutama Hadoop Distributed File System (HDFS).
3. Menginstalasi dan mengkonfigurasi Spark MLlib pada *cluster* Hadoop.
4. Melakukan studi literatur bahasa pemrograman Scala.
5. Melakukan studi literatur tentang algoritma-algoritma pada MLlib dan konsep MLlib.
6. Eksperimen cara pemanggilan fungsi-fungsi pada *library* MLlib.
7. Memodifikasi *source code* algoritma K-Means pada *library* MLlib.
8. Mencari dan mengumpulkan data studi kasus.
9. Merancang dan mengimplementasikan perangkat lunak demo.
10. Melakukan pengujian dan eksperimen untuk menguji performa perangkat lunak untuk menganalisis big data.
11. Menulis dokumen skripsi.

1.6 Sistematika Pembahasan

Bab 1 Pendahuluan

Bab 1 membahas tentang latar belakang, rumusan masalah, tujuan, batasan masalah, metodologi penelitian, dan sistematika pembahasan.

Bab 2 Dasar Teori

Bab 2 membahas tentang teori-teori mengenai sistem terdistribusi, konsep dan cara kerja Apache spark dan hadoop HDFS, perpustakaan Spark MLlib, dan sintaks dasar penulisan dalam bahasa pemrograman scala.

Bab 3 Studi dan Eksplorasi Hadoop

Bab 3 membahas tentang eksperimen-eksperimen dimulai dari langkah-langkah instalasi Apache Spark, eksperimen dengan Spark Shell, dan eksperimen pemanggilan fungsi MLlib.

Bab 4 Analisis

Bab 4 membahas tentang penjabaran kebutuhan pengembangan K-Means MLlib, analisis *source code* K-Means MLlib, dan pengembangan yang dilakukan pada K-Means MLlib sesuai kebutuhan.

Bab 5 Perancangan, Implementasi, dan Pengujian

Bab 5 membahas tentang lingkungan implementasi, implementasi antarmuka perangkat lunak demo, pengujian perangkat lunak demo, dan eksperimen uji performa perangkat lunak demo.

Bab 6 Kesimpulan dan Saran

Bab 6 membahas tentang kesimpulan dan saran.