UNIVERSITÀ DELLA CALABRIA
DIPARTIMENTO DI
INGEGNERIA INFORMATICA,
MODELLISTICA, ELETTRONICA
E SISTEMISTICA
DIMES

Corso di Laurea in

Ingegneria Informatica

Progetto Didattico

Big Data Management

Professori

Sergio Flesca

Studente

Eros De Rose 224482

Sommario

1) Introduzione

1.1) Descrizione e finalità progetto

Il presente documento riguarda la relazione del progetto didattico del corso di Big Data Management tenuto dal professore Sergio Flesca.
In particolare, è stato proposto un progetto riguardante tecnologie viste e spiegate a lezione, quando è stato affrontato il tema del Big Data Processing, e abbiamo visto qual è stata la sua evoluzione, e uno dei modelli di programmazione per analizzare grandi moli di dati è stato il MapReduce, poi ripreso da altri framework, come Apache Spark, molto utilizzato oggigiorno, e che si è scelto come tecnologia di utilizzo per il progetto in questione. Spark e le altre tecnologie utilizzate saranno approfondite nel resto della relazione.
Il progetto proposto è stato quello di gestire e analizzare un dataset contenente i crimini in Italia nel periodo 2016-2020, dove sono presenti diversi campi, tra i quali abbiamo: luogo dove sono stati commessi (regione o città italiane), tipo di delitto, anno in cui è stato commesso, il valore che indica quanti ne sono stati commessi ecc...
La gestione viene effettuata per permettere di realizzare alcune query:

1) Numero totale di crimini per regione in un preciso anno (passato in input)
2) Restituire i delitti di una precisa città (passata in input), in base ad un preciso anno (passato in input), in ordine decrescente
3) Le prime 10 città con il maggior numero di crimini (specificato in input) in un determinato anno (passato in input)

4) Confronto nel periodo 2016-2020, di un crimine (specifico) e di una città (specifica)

L'output di tali query, oltre che localmente, sono stati esportati su MongoDB, un DBMS non relazionale, che permette di visualizzare e salvare le query interrogate, sarà approfondito in seguito.
Le finalità del progetto sono state quelle di capire, imparare ed utilizzare nuove tecnologie, ancora non affrontate durante il percorso di studi, per ampliare le conoscenze su determinati campi, in particolare, la gestione di grandi moli di dati e utilizzare e/o imparare nuovi linguaggi di programmazione come Scala.

1.2) Architettura e tecnologie utilizzate

Di seguito sono elencate le tecnologie utilizzare per la realizzazione del progetto:
Apache Spark: framework open source, è un motore di analisi per l'elaborazione dei dati su larga scala, progettata per essere veloce e affidabile. Fornisce API di alto livello in Java, Scala e Python, e altri strumenti di livello superiore.
Linguaggio Scala: Considerando il framework Apache Spark è stato realizzato in Scala, si è scelto di utilizzare questo linguaggio per due ragioni:

1. Utilizzare lo stesso linguaggio del framework
2. Imparare un nuovo linguaggio di programmazione

MongoDB: è un DBMS, ovvero un Database Management System, progettato per consentire la creazione, manipolazione e interrogazione di database, non relazionale.

2) Sviluppo

3.1) Descrizione generale

È stata sviluppata utilizzando Apache Spark e il linguaggio Scala. Il linguaggio Scala ha evitato la corposità del codice.
Sono state sviluppate quattro classi, equivalenti alle quattro query proposte per il progetto.
L'obiettivo è di analizzare e gestire il dataset di riferimento, a seconda delle richieste che riceve, e fornire i risultati di tale analisi.

3.2) Spark, Scala e MongoDB

Come già accennato in precedenza, è stato utilizzato Apache Spark, un framework realizzato e pensato per la gestione di grandi quantità di dati. Si è scelto di utilizzarlo, anche perché è uno dei sistemi distribuiti più importanti sul mercato oggi. Di seguito alcuni vantaggi di questa tecnologia:

- Velocità: esegue carichi di lavoro 100 volte più rapidamente di Hadoop MapReduce. Spark raggiunge elevate prestazioni sia per i dati in batch che per quelli in flusso.
- Facilità di utilizzo: Spark mette a disposizione più di 80 operatori di alto livello. Si può utilizzare con diversi linguaggi, come Scala, Python, SQL.
- Generalità: Spark supporta uno stack di librerie, tra cui SQL e DataFrames (quest'ultima utilizzata per la gestione del dataset), MLib

per il machine learning. È possibile combinare queste librerie nella stessa applicazione.

Spark è stato utilizzato con il linguaggio di comunicazione Scala.

Scala è un linguaggio di programmazione funzionale ibrido, cioè ha caratteristiche di programmazione orientata agli oggetti e programmazione funzionale. Come linguaggio di programmazione orientata agli oggetti, considera ogni valore come un oggetto, come programmazione funzionale, definisce funzioni anonime, supporta funzioni di ordine superiore e funzioni nidificate.

Classificato come un database di tipo NoSQL, MongoDB si allontana dalla struttura tradizionale basata su tabelle dei database relazionali, rendendo l'integrazione di dati di alcuni tipi di applicazione più facile e veloce.

Di seguito alcuni vantaggi di questa tecnologia:

- Query ad hoc: supporta ricerca per campi, intervalli e regular expression. Le query possono restituire campi specifici del documento.
- Alta Affidabilità: fornisce alta disponibilità e aumento del carico gestito attraverso i replica set, che consiste in due o più copie dei dati. La replica primaria effettua sia scritture che letture, mentre le secondarie mantengono una copia dei dati.
- Bilanciamento dei dati: MongoDB include un meccanismo di bilanciamento dei dati, spostando gli intervalli di dati da uno shard troppo carico a uno shard meno carico, in modo da bilanciare la distribuzione dei dati all'interno del cluster.
- File Storage: può essere usato anche come un file system, traendo vantaggio dalle caratteristiche di replicazione e di bilanciamento su più server per memorizzare file, anche di grandi dimensioni.

3.3) Filtrare le richieste

Per poter operare sulle quattro query, sono state create quattro classi scala, che operano in maniera simile tra di loro, ma che vanno a filtrare le diverse richieste. In particolare, per la prima query, bisogna applicare tre filtri, uno che vada a prendere le righe relative alle regioni escludendo le altre, un altro che prende il totale dei crimini e per ultimo un filtro che seleziona l'anno passato in input al metodo. Dopodiché nel nuovo dataframe creato filtrando le richieste, si vanno a selezionare solo le colonne che si vogliono visualizzare in output e che saranno visualizzate su MongoDB.
Infine, prima del salvataggio, si ordinano le righe del nuovo dataframe in ordine decrescente in base al valore dei crimini.
Per il salvataggio, il nuovo dataframe creato viene scritto e salvato su MongoDB, in modo da poter visualizzare il risultato.
Lavorando in modo simile, vengono filtrate anche le altre query, in particolare, nella seconda si applicano due filtri, che si riferiscono al luogo e all'anno dei crimini e poi, come spiegato per la prima query, si selezionano le colonne interessate, da restituire, e si ordinano anche in questo caso per numero di crimini in ordine decrescente e infine, il nuovo dataframe filtrato, viene salvato su mongoDB.
Per la terza, i filtri sono applicati sull'anno e sul tipo di delitto, entrambi passati in input, e viene utilizzata una particolare funzione che seleziona solo le città presenti nel file csv, escludendo le regioni e altre righe indesiderate.
Per la quarta, i filtri sono applicati sul tipo di delitto e sulla città desiderata, entrambi passati in input. Questa volta però l'ordine è ascendente in base all'anno, poiché si vuole fare un confronto del periodo 2016-2020.
Il linguaggio Scala ha permesso di ridurre la quantità di codice, rispetto al linguaggio Java, infatti, tutti i controlli nei metodi sono stati effettuati

utilizzando poche righe di codice, sfruttando la libreria DataFrames, che ha al suo interno diverse funzioni per gestire velocemente un dataset.

3.4) Output

Per l'esportazione dell'output è stato scelto, come già accennato, MongoDB, che può essere usato attraverso il terminale, ma è stata rilasciata anche una GUI, che permetta la visualizzazione dei dati più "user-friendly", e quest'ultima è stata utilizzata per il progetto del corso.
L'immagine sottostante permette di visualizzare come appare l'output di una query lanciata, in particolare la quarta, dove si confrontano i dati di uno specifico crimine in una determinata città, e per l'esempio sono stati inseriti come città, "Cosenza", e come crimine "omicidi colposi":