TERM PROJECT

MET CS 779 Term Project

Abstract

The term project implements a DBLP Paper Retrieval System Using Hadoop. The goal of the project is to provide a user-friendly local interface for users to search the papers using a reasonable ranking method.

Chuqian Zeng

cqzeng@bu.edu

MET CS 779 Assignment

Table of Contents

1. Introduction

The term project is called A DBLP Paper Retrieval System Using Hadoop. The goal of the project is to provide a user-friendly local interface for users to search the papers using a reasonable ranking method. The techniques and tools used in the project include MySQL, Hadoop, Java, HTML and information retrieval algorithm.

2. Background

2.1 Hadoop

The Apache Hadoop software library is a framework that allows for the distributed processing of large data sets across clusters of computers using simple programming models (Apache, 2021). The MapReduce it provided is a YARN-based system for parallel processing of large data sets, which is extremely suitable for building the inverted index we need in this project.

2.2 Information Retrieval Algorithm

We use inverted index (Wiki.Inverted_index, 2021) and TF-IDF (Wiki.TF_IDF, 2021) to generate our retrieval algorithm. They are widely used in information retrieval.

3. Methodology

3.1 Installation

The source code is provided with this report, with a project name "DBLP Retrieval". Following the steps below to re-build the project.

- Install WAMPServer as our database management system.

WampServer is a Windows web development environment which aggregates Apache, PHP and MySQL. In our case we use its MySQL part as our database management system.

- Create database.

Run the dblp_script.sql script located in DBLP Retrieval/src/script/ in the WampServer to create the database.

- Build Project.

Java should be implemented in the local environment. In our case we use Eclipse to build our project. Launch our project into Eclipse. Download the JDBC package and put it in DBLP Retrieval/lib, and import the package into the project build path.

- Implement Hadoop.

Follow the instruction to implement Hadoop in our local environment and run Hadoop. In our case, we use Hadoop 2.9.1. Also import all the JAR packages inside the folders and subfolders in %HADOOP_HOME%/share/Hadoop into the project build path.

- Obtain necessary data files.

Download dblp.xml.gz and dblp.dtd from DBLP source. The dblp.xml should be extracted from the dblp.xml.gz compressed file. Put the dblp.xml and dblp.dtd into DBLP Retrieval/data.

- Insert data to WAMP.

Change the user and password in DBLP Retrieval/src/connect_database/Connector.java to your own username and password for the database. If WAMP is installed for the first time, the default username should be "root" and password should be empty. Then compile and run DBLP Retrieval/src/make_database/MyParser.java to extract information from the XML file and insert them into the database.

- Generate inverted index using Hadoop.

Make sure Hadoop is successfully imported into the project. Then compile and run DBLP Retrieval/src/map_reduce/MapReduceExecutor.java to generate the inverted index files. The input data for this MapReduce process are the data in the database that we generated from the previous step. The output data are placed in the path DBLP Retrieval/data.

- Run local server.

Compile and run DBLP Retrieval/src/Interface/UserInterface.java to start the local server.

- Use website to query.

Open DBLP Retrieval/src/index.html in the browser. Now you can search query using this Google-style website.

3.2 Functions

Our project includes the following functions:

- Query for Paper.

Supports basic user interaction, such as entering a query string and returning results (based on paper topic information).

- Domain Retrieval.

Domain retrieval is supported for domains such as titles, authors, and journals.

- Paper Details.

Supports user-friendly interaction: users can see more information when they click the link of the returned results.

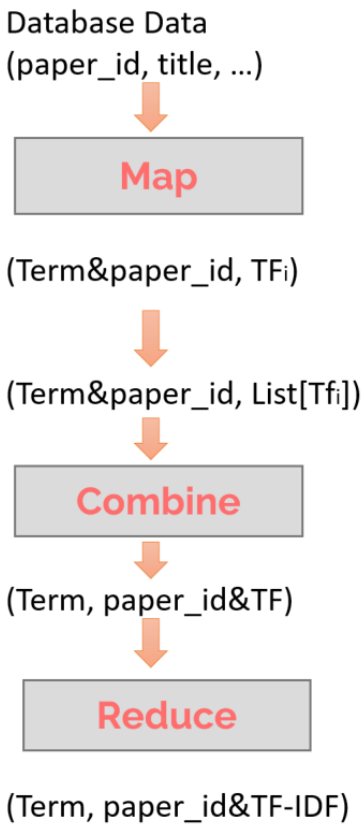
- Timeliness of Paper.

The timeliness of the paper is taken into account when sorting the returned results (based on the year of publication) by adding attenuation of year.

3.3 Theory

The most important part of the project is the Hadoop MapReduce, which is used to generate the inverted index.

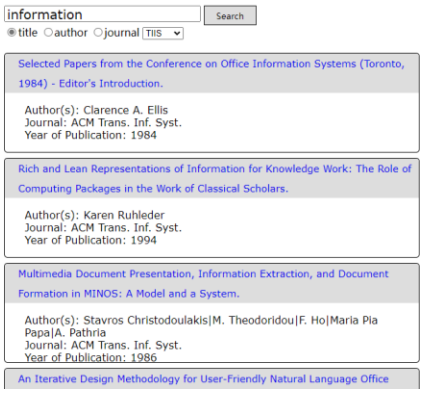
Our Hadoop program includes three parts, Map, Combine and Reduce. The Map procedure takes in a row in MySQL database, and output a pair with key of term and paper id, and value of term frequency for one word. The Combine procedure takes in a specific term and a paper id as a key, and a list of corresponding term frequency as values. The output of the Combine procedure is a term as key, and a paper id with the corresponding term frequency for this term in the paper as value. The Reduce procedure takes in a term as a key, and a list of corresponding paper id with term frequency as value, and output the term as key, and paper id with TF-IDF value as value. After the MapReduce procedure, we generate the TF-IDF value for each term in each document, which we can use to easily create the inverted index.



As we have generated the inverted index, each document can be represented by a vector whose elements are the word dictionary, and the weight of each element is the TF-IDF value of the word (call "term") in the document. Then we can find the most matched documents to the query by calculating the cosine similarity of the vector of the query and that of each document.

4. Result

Standard searching with title.



Searching with title and author.

Searching with journal:

5. What I learn

5.1 From the project

The project is based on one of my previous projects about information retrieval. The older version just loaded data from the database and made calculation in the memory. In this update version, I implement Hadoop MapReduce for the first time to generate the inverted index, which is stored in storage and accelerates the querying process. I also update the older JSP front-end to more standard HTML using AJAX to send and receive messages.

The implementation of Hadoop was frustrating. In the beginning I planned to install the latest version of Hadoop. However, because of some conflicts with Windows, the latest Hadoop 3.3.1 did not work on my computer environment; it kept showing error when I tried to start Hadoop. I searched about the error message for quite a long while and made no outcome. Finally, I switched to older 2.10.1 version and made it work. Another problem I encountered laid in inputting data from database to MapReduce. I was first using SQL Server as my database. However, when executing the pre-defined SQL command, it showed an error because of the difference of “limit” and “top” syntax in MySQL and SQL Server. I could not fix that because I had no way to change the SQL command inside the JAR package. Finally, I switched to MySQL to make it work. These are the two problems that took away most of my time when implementing Hadoop, not to mention all the small bugs when executing.

Although experiencing a hard time, I finally came out with a successfully running program with something new in it. Through this learning experience I became more familiar with Hadoop, and developed the skills to solve problems when implementing new environment.

5.2 From the course

I really learned a lot from this course. I previously had some basic knowledge about SQL command, and this course led me to go deeper, by implementing more complex commands through programming, and more importantly comprehending the design methods of database from a real Netflix case. The practical design of assignments and homework guided me to progress pace to pace.

6. Conclusion

The term project presents a complete application of information retrieval system of DBLP paper leveraging Hadoop. In the future, we are planning to develop a distribute version to support access to multiple servers.

5. Revision History

Name	Date	Version	Description
Chuqian Zeng	12/13/21	1.0	Initial Document Creation
Chuqian Zeng	12/13/21	1.1	Final Version

Appendices

Source code

A project DBLP Retrieval

PowerPoint

Zeng_Chuiqn_term_project.pptx

Bibliography

Apache. (2021). Retrieved from Apache Hadoop: https://hadoop.apache.org/

Wiki.Inverted_index. (2021). *https://en.wikipedia.org/wiki/Inverted_index*. Retrieved from Inverted index.

Wiki.TF_IDF. (2021). Retrieved from TF-IDF: https://en.wikipedia.org/wiki/Tf%E2%80%93idf