Project Report

CS 453: Project 4

Distributed Query Processing

Larry Bowers
killergift@wsu.edu

James Bradwell
jebradwell@gmail.com

Evan Dickinson
evan13579b@wsu.edu

Bhadresh Patel
bhadresh@wsu.edu

Lewis Pearson
lewis_pearson@wsu.edu

Washington State University Vancouver
December 05, 2010

Abstract

The main goal of this project is to provide user interface for the search engine and process user's query over multiple Amazon EC2 nodes using the PageRank and Indexes generated by previous project.

1 Overview

2 Interface

2.1 Query Input

2.2 Result Output

3 Distributed Evaluation

3.1 Distributer

3.2 Index Server

3.3 Merge Results

4 Index Compression

5 Retrieval Model

5.1 Query Likelihood

5.2 BM25

6 Roles

Larry Bowers User interface and query parser

James Bradwell User interface and query parser

Evan Dickinson Huffman encoding/decoding

Bhadresh Patel Distributed evaluation

Lewis Pearson Retrieval model

7 Test Environment

For testing/production purpose, we have set up instances on Amazon EC2. The instance id of the director machine is i-5135773c which also hosts the user interface. Instance ids of five index servers are: (i) i-5335773e, (ii) i-2d357740, (iii) i-2f357742, (iv) i-29357744, and (v) i-2b357746. The source code on all instances is checked out at /home/ubuntu/dqp/. The user interface can be accessed via public DNS name of the director instance. For example, http://ec2-174-129-159-24.compute-1.amazonaws.com/ui/.

8 Usage Guide

- Update Nodes file `dp/local.nodes` or `dp/cloud.nodes`. Change the IP address of all nodes for the appropriate environment.
- Start index server on each instance

  ```
  $ python ~/dqp/dp/server.py
  ```

- Find the public DNS of the director instance `i-5135773c`. Launch the UI using public DNS, for example, http://ec2-174-129-159-24.compute-1.amazonaws.com/ui/.

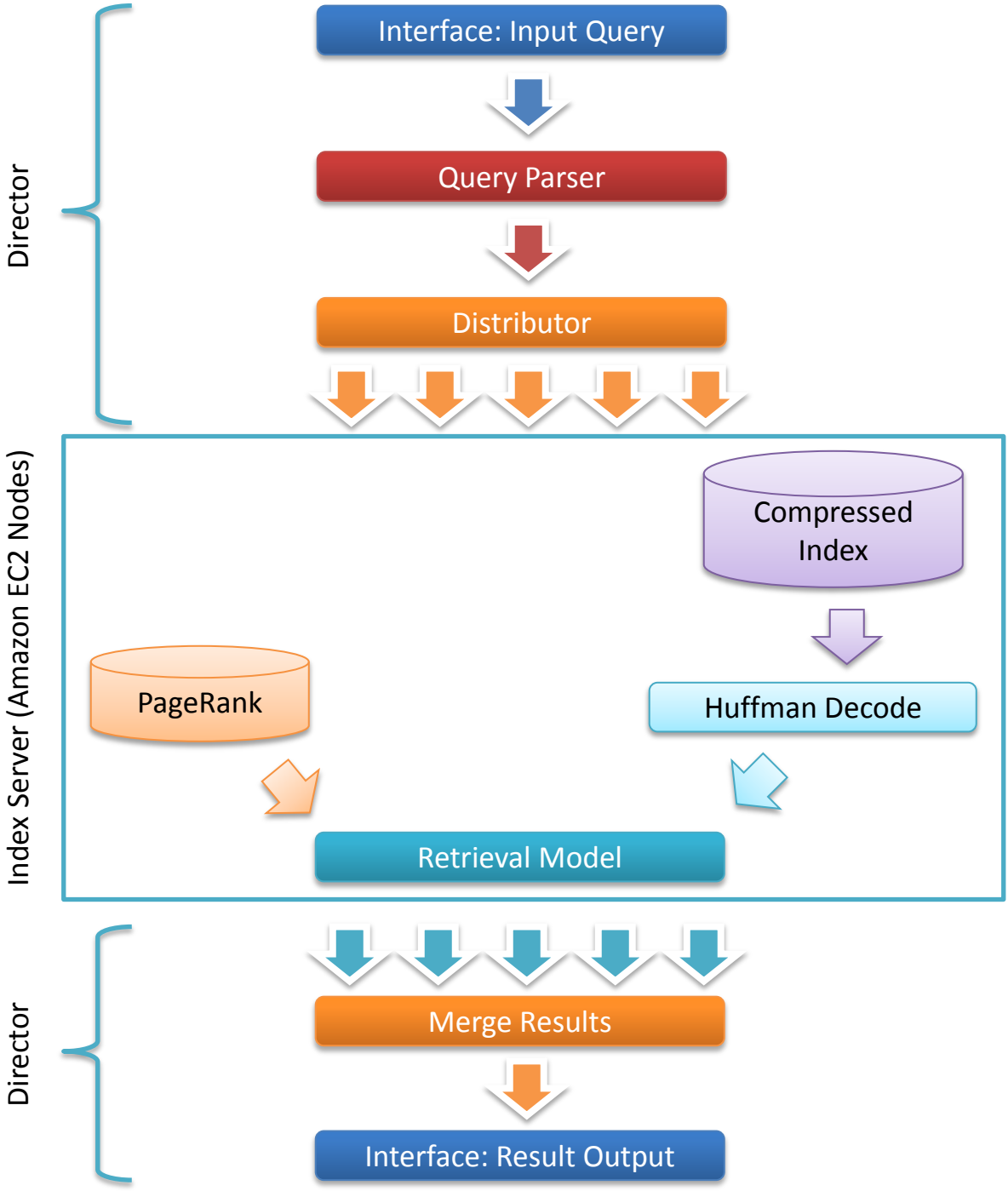

Figure 1: Architecture Overview