UNIVERSITETET I AGDER

Project report in Dat300

Distributed querying with Apache Solr

Improving performance by splitting complex search expressions

A bachelor thesis by

Mari Næss
Ørjan Hatteberg
Enok Karlsen Eskeland

Supervisors

Folke Haugland
Jaran Nilsen

This Bachelor's Thesis is carried out as a part of the education at the University of Agder and is therefore approved as a part of this education.

University of Agder, 2011

Faculty of Engineering and Science

Department of information- and communication technology

Abstract

Complex taxonomies delivers bad search performance for Integrasco. This report is about troubleshooting the issue and developing a solution based on a hypothesis stating that several smaller taxonomies in sum performs better than one large taxonomy. Testing indicated optimization potential in splitting. This sparked the creation of a query splitter to decrease response time in a sharded environment on Solr. This splitter proved to give improved performance when a taxonomy performs poorly with the regular search. Despite a problem relating to searches with high start offset, this can be a beneficial solution for the problem.

Contents

1 Introduction **5**

2 Theory **6**

3 Solution **7**

3.1 Fire Interpreter 7

4 Discussion **8**

5 Conclusion **9**

Acknowledgments **10**

References **11**

List of Figures

List of Tables

Definition list

Taxonomy is by Integrasco usage and in this context defined as a complex query.

Document is the basic unit in Lucene indexing. E.g. a single pdf or a book.

Rows is the number of documents in the result set of a query.

Start Offset is the index of the first document you want displayed.

Page Offset is used in pagination, but is the same as start offset.

Iterations are the number of times a taxonomy is queried.

Hit Count is the total number of documents matching the query.

QueryOptimizer library is the solution developed for the problem.

QTime is the time spent generating the in memory response for a query in Solr (milliseconds).

Elapsed Time is QTime plus serializing and de-serializing transmitting in Solr (milliseconds).

Query Time is the time it takes to perform a solr search from QueryOptimizer or the test framework (milliseconds).

Lucene is an open source free text search library from Apache.

Solr is an open source search server utilizing Lucene.

Solrconfig.xml contains the parameters to configure Solr.

QueryResultWindowSize . A window is a section of search results. It can be from 0-49, 50-99 etc. When querying the entire window in which the search match will be returned and loaded into cache. QueryResultWindowSize is the size of these windows.

QueryResultMaxDocsCached is the maximum number of documents a single query can have in cache memory.

Index is a sorted list of terms present in the data set. Contains links for finding the term locations.

Sharded index is an index split in smaller parts possibly on different servers to better cope with scaling issues.

Chapter 1

Introduction

Chapter 2

Theory

Chapter 3

Solution

3.1 Fire Interpreter

The fire interpreter's job is to receive input from the simulator and calculate values for the received cells. The calculated values are then posted to the visualizer. The advanced calculations are done by the library pyXGPR. This is a Gaussian Process Regression library implemented with Python. It produces a mean and a variance when used correctly. The first input parameter **X** is a list of points which tells where the training data is located. Another parameter **Y** contains the values to the training data. The last interpreter generated parameter **x star** contains the points where we want to find the mean and the variance. In addition to these parameters the library needs to be told what covariance functions pyXGPR should use to calculate the correlation between the cells in **X**, **Y** and **x star**. There is also added parameter values to these functions.

The most basic use of pyXGPR is one dimensional (line regression) where **X** is the location and **Y** is the value. The interpreter uses **X** and **Y** for the map coordinate and an additional parameter **t** for time. **t** is necessary to save earlier sensor data which later are utilized in calculations. It should also be mentioned that before this implementation, this was done by saving the best data. Best data is to be understood as the data which has the lowest variance. Data with lower variance would be applied to the saved map. This hack and the implementation of t is done because previous sensor data is important as long as they are weighted less than the newest sensor data. As time increases there will be sensor data covering most of the map, but the old sensor data will have less weight and thus giving new sensor data the opportunity to be taken into account.

Chapter 4

Discussion

Chapter 5

Conclusion

Acknowledgments

We would like to thank our supervisors Folke Haugland and Jaran Nilsen for their constructive feedback that has led to progress in times when the project was at a stand still. We would also like to thank Integrasco for letting us use their office to store the server and work in, and University of Agder for lending us the server.

02 june 2011
University of Agder

Bibliography

[1] Smiley, D & Pugh, E. (2009). *Solr 1.4 Enterprise Search Server.* Birmingham UK: PACKT publishing.

[2] McCandless, M & Hatcher, E & Gospodnetic, O. (2010). *Lucene in action (2th ed.).* Stamford, CT: Manning Publications Co.

[3] *Apache Solr Wiki* available at `http://wiki.apache.org/solr/FrontPage`

[4] *Apache solrconfig.xml* available at `http://svn.apache.org/repos/asf/lucene/dev/trunk/solr/example/solr/conf/solrconfig.xml`

[5] Mak, Gary; Long, Josh; Rubio, Daniel. *Spring Recipes (2010).* Apress

[6] Astels, David. *Test-Driven Development: A Practical Guide (2003).* Prentice Hall

[7] O'Brien, Tim; van Zyl, Jason; Fox, Brian; Casey, John; Xu, Juven; Locher, Thomas; Moser, Manfred *Maven:The Complete Reference (2010)*