WDM - Project Proposal - Group Project-1

DATASET

During this project we will be using the SNAP Wikipedia dataset, which contains the complete edit history up to January 2008. We will focus on the user talk pages and user personal pages edit history, as well as the Wiki namespace which contains administrative procedures and pages.

SYSTEMS

PostgreSQL

PostgreSQL is the second most-used open-source DBMS, after MySQL. PostgreSQL has a high SQL standard compliance, A benchmark provides the information that PostgreSQL is faster in aggregation and write-only queries. This is useful for this dataset, as in most cases the query will be an INSERT. PostgreSQL is slow when updating a bulk at ones, as the wiki talk data is historical data this will not be that big a deal. Since PostgreSQL version 9.6 it supports parallel sequential scan, which is the first step towards parallel queries. Parallel queering can significantly speed-up aggregations and joins, this can be useful when we want to query the amount of edits group by user, type and date for analytical purpose.

Neo4j

Neo4j is one of the most-used graph based databases. Graph databases provides much better performance when it comes to querying deeply connected data that has many relationships expressed with complex joins compared to traditional relational databases. Such a system can be beneficial if we want to query users which can potentially help writing an article by finding users which the users you talked discussed with in other talk pages on similar articles. A limitation of Neo4j is the exclusion of range indexes, which can make certain operations such as sorting quite expensive. This can be problematic when we want to query wiki talk pages sorted by activity.

Apache Lucene/ElasticSearch

Apache Lucene is based on Java and is geared towards search and indexing, while delivering a lot more related functionality. Lucene has the capability to do full text indexing and searching but is renowned for its ease of being the core of a search engine. ElasticSearch is one of those systems with Lucene at its core, it is a distributed NoSQL database, integrating a high-performance search and analytics system, this is beneficial when, for example, searching for users with a specific (sub)string in their name (to possibly detect bots). The system should be very scalable, perform good on search queries and should be easy to setup due to the schemaless nature. However, ElasticSearch was never meant to be used as a primary database and still has reliability/consistency issues.

Apache Spark

Apache Spark is open-source cluster-computing framework, maintained by the Apache Software Foundation. Its features are a resilient distributed dataset, distributed over multiple machines which can be maintained in a fault-tolerant way. Queries are implicitly parallelised by it cluster based setup. Spark is setup using a cluster manager and a distributed storage system, which both can be implemented using a wide variety of techniques / languages. We expect this system to work well on the big dataset, however due to the iterative setup, having a lot of joins will mean iterating a lot over the complete dataset, for example finding 2nd degree people with similar interests by pages a user discussed on.

EVALUATION

We will use the following metrics in order to evaluate the different systems: **Query duration**. **Scalability**, how easy is it to scale the system and how does it increase performance. **User friendliness**, how complex is an query in order to perform a task **Setup time/cost,** how long does it take to import data and setup the database **Memory**, how large is the database created **Query throughput**

MILESTONES

- May 13 Setup different systems
- May 15 Feedback proposal
- May 20 Implementation of queries
- May 27 Finish analysis of queries / benchmarking
- June 1 Finish report
- June 5 Deadline report
- June 11 Presentations

ACM ISBN 978-1-4503-2138-9.
DOI: 10.1145/1235