Pivoting Entity-Attribute-Value data Using MapReduce for Bulk Extraction

Augustus Kamau

Faculty Member,

Computer Science, DeKUT

Overview

- Entity-Attribute-Value (EAV) data
- Pivoting
- MapReduce
- Experiment & Results
- What Next
- Q & A Time

Entity-Attribute-Value (EAV) Data

- Entity Attribute Value (EAV) is vertical data modelling, as opposed to conventional horizontal data modelling, made up of at least three columns:
  - Entity: the object being described
  - Attribute: object properties
  - Value: the value of an attribute
  - Also known as: object–attribute–value, Vertical DB, Open Schema, tall/skinny schema
- Suitable for storing rapidly evolving and sparsely populated data. Example applications: E-commerce Applications, Medical Information Systems and Recommender Systems

Horizontal Vs Vertical Representation

Entity	A1	A2	A3	A4	A5
E1	V1			V2	
E2		V3			V4
E3			V5	V6	
E4		V7			
E5			V8	V9	V10

Horizontal

Entity	Attribute	Value
E1	A1	V1
E1	A4	V2
E2	A2	V3
E2	A5	V4
E3	A3	V5
E3	A4	V6
E4	A2	V7
E5	A3	V8
E5	A4	V9
E5	A5	V10

Vertical (EAV)

- Because most current applications and development tools are designed for horizontal format, reconstruct (pivot) from vertical representation to horizontal table is often necessary.

Pivoting

- Pivoting is an operation on tabular data that exchange rows and columns, enable data transformations useful in data modelling, data analysis, and data presentation (Cunningham et al., 2004)
- Data in one source column is used to determine the new column for a row, and another source column is used as the data for that new column.

Entity	Attribute	Value
E1	A1	V1
E1	A4	V2
E2	A2	V3
E2	A5	V4
E3	A3	V5
E3	A4	V6
E4	A2	V7
E5	A3	V8
E5	A4	V9
E5	A5	V10

Pivot

Entity	A1	A2	A3	A4	A5
E1	V1			V2	
E2		V3			V4
E3			V5	V6	
E4		V7			
E5			V8	V9	V10

Pivoting Operation

- Pivoting operation is currently provided in various software applications such as spreadsheets e.g., Ms. Excel, Commercial DBMSs e.g., Oracle and SQL Server, and some analytical tools such as XLSTAT.
- Pivoting is a time-consuming and resource-intensive process and is often performed repeatedly on regular basis.
- Thus it would be beneficial to make pivot operations as efficient as possible

Pivoting Approaches

- They are various pivoting approaches being used today:
  - Using Structured Query Language (SQL) joins statements and set operations
  - PIVOT relational operator.
  - Materialized views
  - Use of hash tables
- The above approaches are optimized for relatively small datasets and therefore for bulky data an efficient approach is needed. MapReduce (Dean and Ghemawat, 2004) which was popularized by google can be used to achieve this.

MapReduce

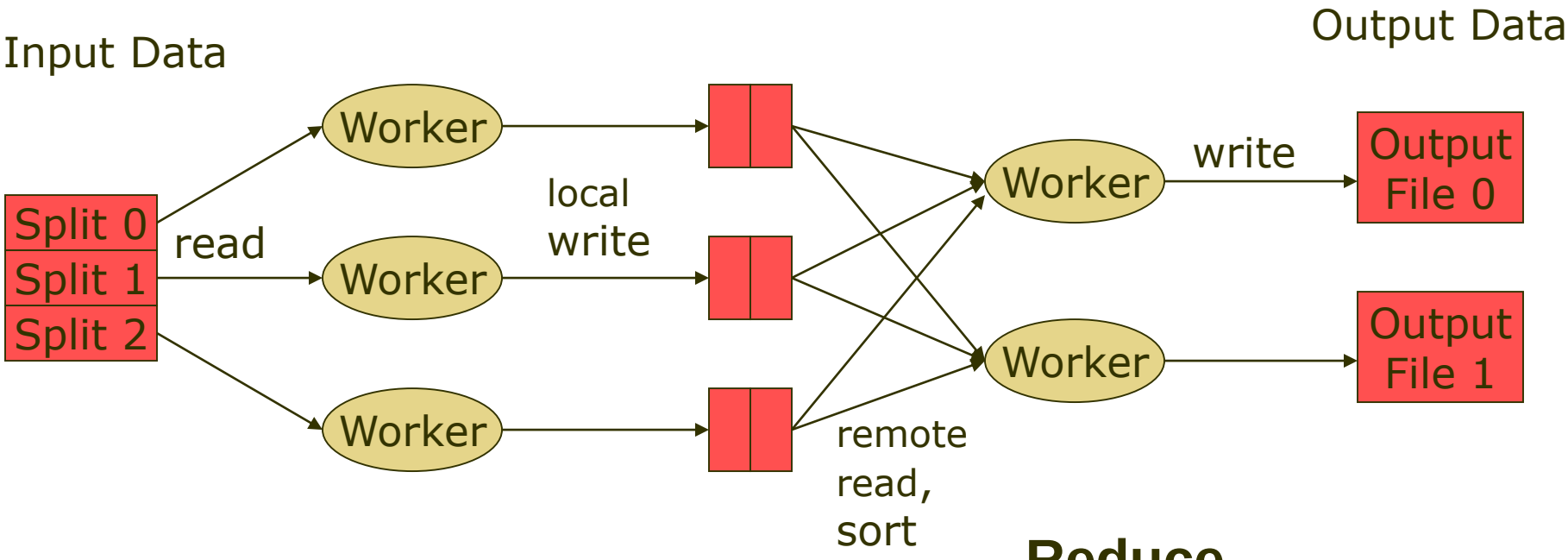
- MapReduce provides
  - Automatic parallelization, distribution
  - I/O scheduling
    - Load balancing
  - Network and data transfer optimization
- Fault tolerance
  - Handling of machine failures
- Need more power: Scale out, not up!
  - Large number of commodity servers as opposed to some high end specialized servers

Typical problem solved by MapReduce

- Read a lot of data
- Map: extract something you care about from each record
- Shuffle and Sort
- Reduce: aggregate, summarize, filter, or transform
- Write the results

- Functions:
  - map (in_key, in_value) -> (out_key, intermediate_value) list
  - reduce (out_key, intermediate_value list) ->out_value list

MapReduce workflow



Map
extract something you care about from each record

Reduce
aggregate, summarize, filter, or transform

Pivoting EAV data using MapReduce

- Map side:
  - Read the records of EAV data from source file line by line for entity, attribute, and value.
  - Emit the entity as the key and the attribute and value pair as the intermediate value.
- Reduce side:
  - Receive and iterate over the passed values for each key
  - At each inner iteration hold the values for each key in a one-dimensional array with the position of each value being determined by the corresponding attribute.
  - Convert the array into a formatted string and emit the result for writing to a file with the row being determined by the key.

Implementation and Experiment Setups

- Three algorithms were implemented in Java (the MapReduce solution was implemented using Hadoop – based on Java).
- Method A:
  - Implemented according to the algorithm given in (Dinu et al., 2006) using hash tables, two-dimensional array, and data from database;
- Method A':
  - Modification of method A to closely correspond to method B in terms of input source (text file), and output (text file) for fair comparison
- Method B:
  - Implemented the algorithm above for pivoting using MapReduce
- The functionalities of the algorithms were tested using Junit, MRUnit, and Mockito testing framework for unit testing and integration testing.

Implementation and Experiment Setups

- Data set from acupuncture headache trial (Victers, 2006) in XLS format.
  - The data is made up of 93 attributes and 401 rows
- Data was synthetically grown to increase the number of rows and split into files based on the number of entities.
- Sample description

S/N	No. of Entities	Data Points	Size (MB)
1.	10,000	737,143	8.36
2.	20,000	1,474,263	17.5
3.	30,000	2,211,432	26.6
4.	40,000	2,948,573	35.7
5.	50,000	3,685,787	44.9
6.	60,000	4,422,788	54.0
7.	70,000	5,160,063	63.2
8.	80,000	5,897,241	72.3
9.	90,000	6,634,152	81.5
10.	100,000	7,371,431	90.6

Implementation and Experiment Setups

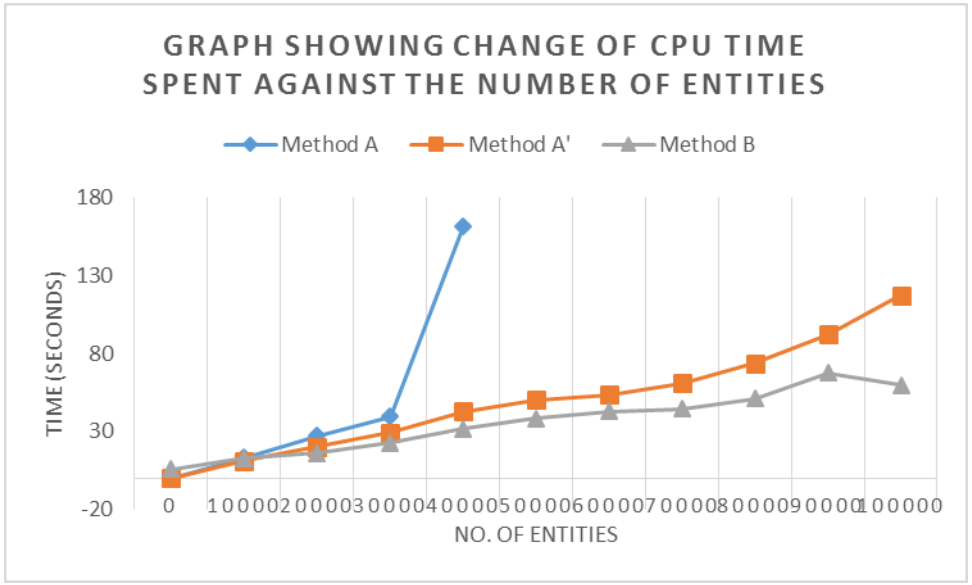
- The programs were run on the same dedicated virtual machine installed with centos 64-bit as guset o/s with the variation of the CPU and Memory allocation as follows:

	No. of CPUs @ 2.10 GHz	Memory (GB)
Exp 1	2	3
Exp 2	2	4.5
Exp 3	4	3
Exp 4	4	4.5

- Each test was run at least three times and averaged the results.
- The CPU time and memory consumed by each run was obtained using YourKit (profiling tool for java) attached to eclipse IDE
- Independent variable:
  - no. of entities
- Dependent variables:
  - CPU time and memory spent

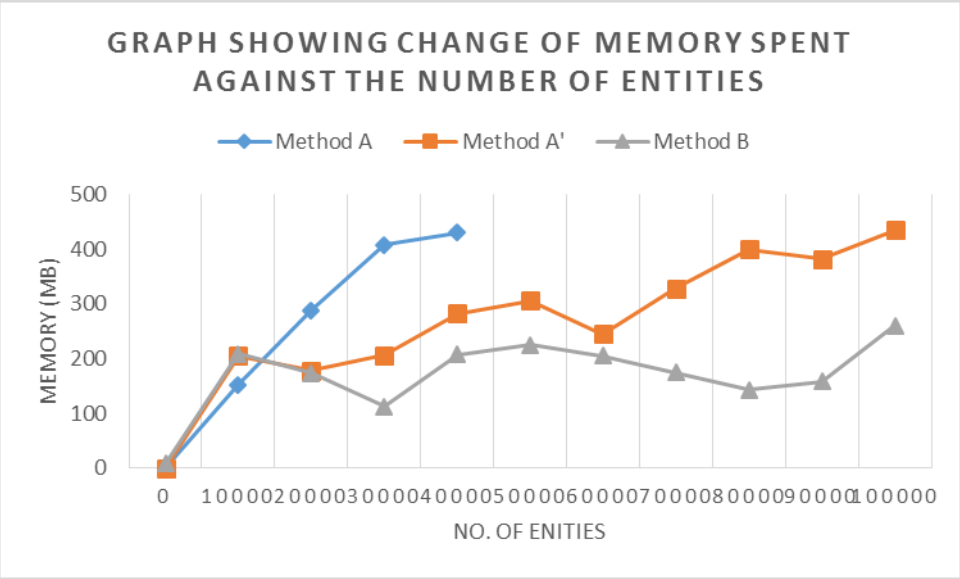
Experiment results and Discussions

- Exp1: 2 CPUs @ 2.1 GHz, Memory 3 GB



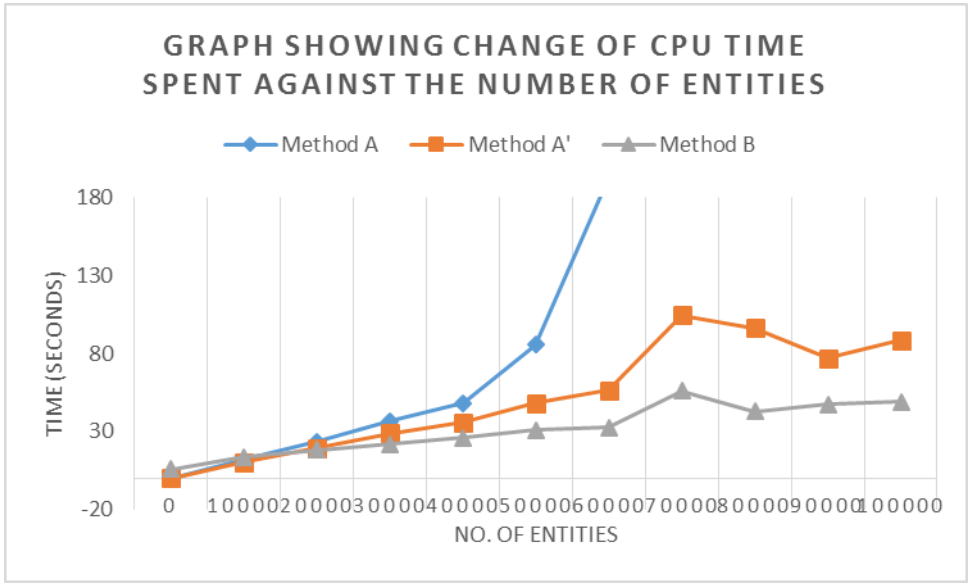
Experiment results and Discussions

- Exp1: 2 CPUs @ 2.1 GHz, Memory 3 GB



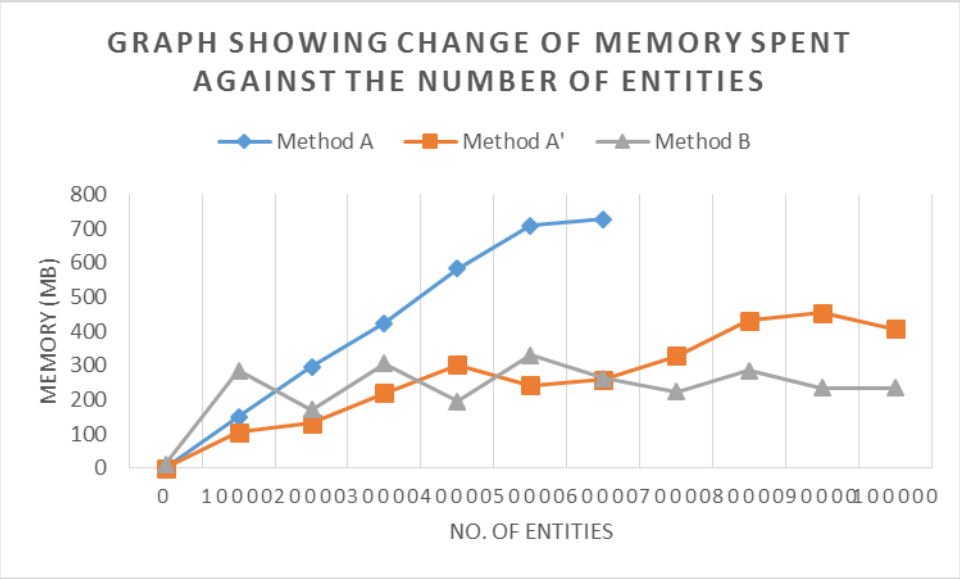
Experiment results and Discussions

- Exp2: 2 CPUs @ 2.1 GHz, Memory 4.5 GB



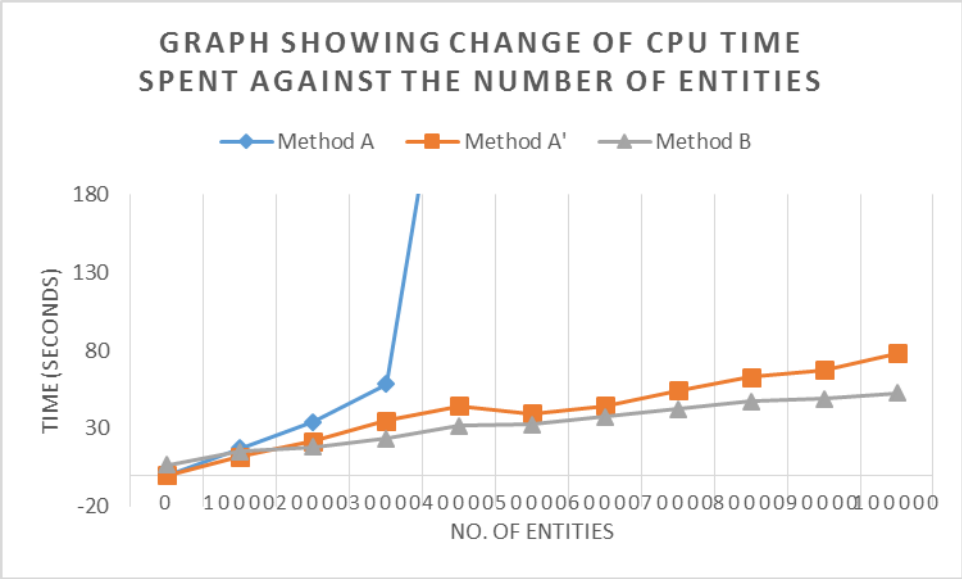
Experiment results and Discussions

- Exp2: 2 CPUs @ 2.1 GHz, Memory 4.5 GB



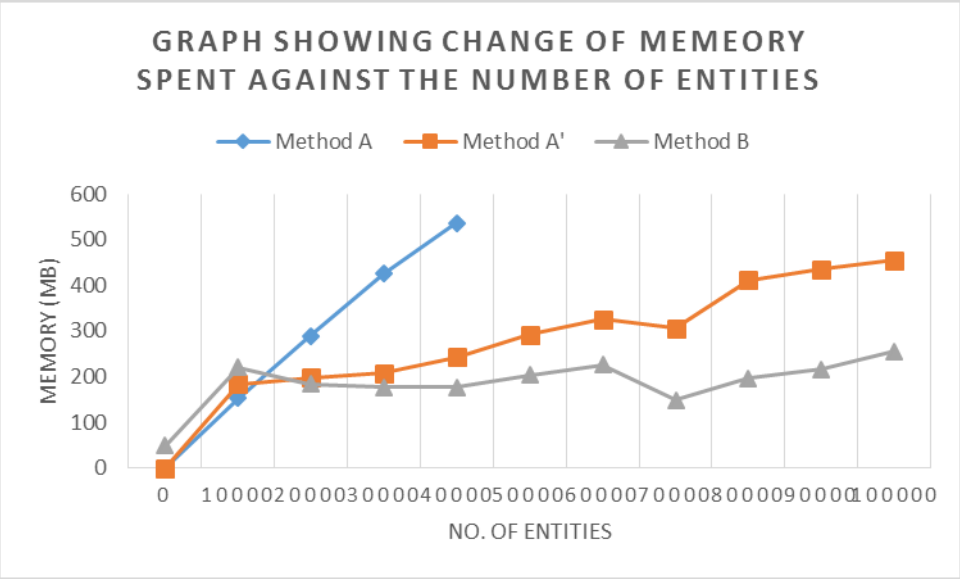
Experiment results and Discussions

- Exp3: 4 CPUs @ 2.1 GHz, Memory 3 GB



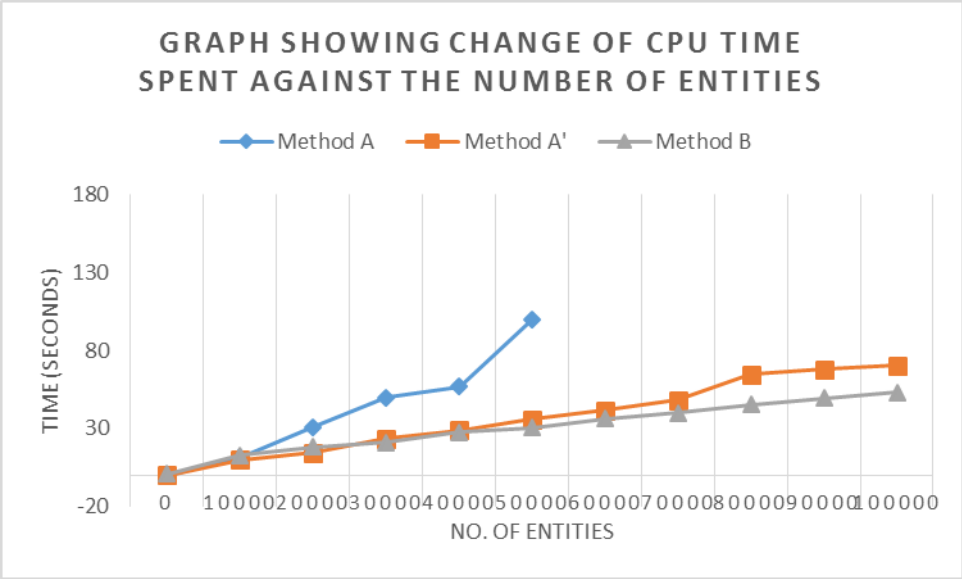
Experiment results and Discussions

- Exp3: 4 CPUs @ 2.1 GHz, Memory 3 GB



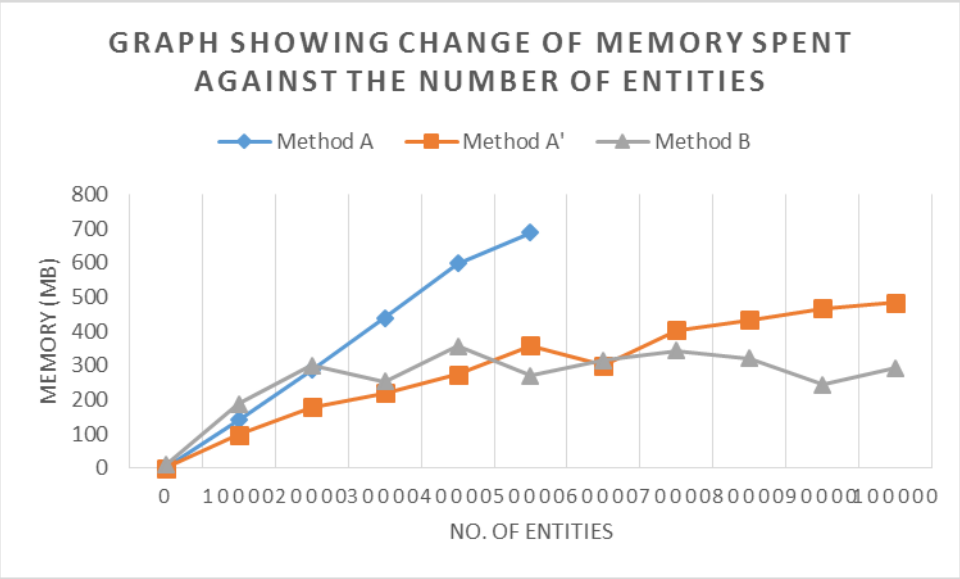
Experiment results and Discussions

- Exp4: 4 CPUs @ 2.1 GHz, Memory 4.5 GB



Experiment results and Discussions

- Exp4: 4 CPUs @ 2.1 GHz, Memory 4.5 GB



Whats next?

- My next steps:
  - Optimize the MapReduce approach presented here using in-memory execution
  - Build a data extraction tool to simplify the specification of and improve the efficiency of extracting data from the EAV model.

References

- J. Corwin, A. Silberschatz, P. L. Miller, and L. Marenco, "Dynamic tables: an architecture for managing evolving, heterogeneous biomedical data in relational database management systems," J. Am. Med. Inform. Assoc., vol. 14, no. 1, pp. 86-93, 2007.
- R. Agrawal, A. Somani, and Y. Xu, "Storage and querying of Ecommerce data," in Proc. VLDB'01, 2001, pp. 149-158.
- J. Gilchrist, M. Frize, C. M. Ennett, E. Bariciak, "Performance evaluation of various storage formats for clinical data repositories," Instrumentation and Measurement, IEEE Transactions on, vol. 60, no. 10, pp. 3244-3252, Oct. 2011.
- V. Dinu and P. M. Nadkarni, "Guidelines for the effective use of entity-attribute-value modeling for biomedical databases," I. J. Medical Informatics, vol. 76, no. 11-12, pp. 769-779, 2007.
- C. Cunningham, G. Graefe, and C. A. Galindo-Legaria, "PIVOT and UNPIVOT: optimization and execution strategies in an RDBMS," in Proc. VLDB'04, 2004, pp. 998-1009.
- J. Dean and S. Ghemawat,"Mapreduce: Simplified data processing on large clusters. In OSDI" 04: Sixth Symposium on Operating System Design and Implementation, San Francisco, CA, 2004.

References

- R. L¨ammel. "Google's mapreduce programming model — revisited," Science of Computer Programming, 68(3):208–237, 2007.
- L. V. S. Lakshmanan, F. Sadri, and S. N. Subramanian, "On efficiently implementing SchemaSQL on an SQL database system," in Proc. VLDB'99, 1999, pp. 471-482.
- P. M. Nadkarni and C. Brandt, "Data extraction and ad hoc query of an entity-attribute-value database," J. Am. Med. Inform. Assoc., vol. 5, no. 6, pp. 511-27, 1998.
- J. Anhoj "Generic design of web-based clinical databases", J. Med. Internet Res., vol. 5, no. e27, 2003
- V. Dinu, P. M. Nadkarni, and C.Brandt, "Pivoting approaches for bulk extraction of entity-attribute-value data," Computer Methods and Programs in Biomedicine, vol. 82, no. 1, pp. 38-43, 2006.
- Apache hadoop. http://hadoop.apache.org/core/.
- A.J. Vickers , R.W. Rees ,C.E. Zollman ,R. McCarney , C.M. Smith , N. Ellis , P. Fisher ,R.V. Haselen, "Acupuncture for chronic headache in primary care: large, pragmatic, randomized trial," Bmj2004, 328(7442):744.
- [15] YourKit, https://www.yourkit.com/.

Q & A Time

- Ask your question
- Receive your answer
- Wash, rinse, repeat

Thanks!

Augustus Kamau

Augustus.Kamau@dkut.ac.ke