Do the middle letters of "OLAP" stand for Linear Algebra ("LA")?

Speaker: Luís A. Bastião Silva
Paper authors: Hugo Daniel Macedo and José Nuno Oliveira
Doctoral Program

MAPi

Summary

- Motivation
- Goals
- Background
- Cross tabulations in LA
- Higher-dimensional OLAP
- Conclusion and future work

Motivation

- Nowadays, companies are creating a huge amount of data
- Big data trend
- They need to access to the information stored in these databases and calculate some metrics
- OLAP (Online Analytical Processing):
    - Summarize huge amount of information
    - Forms of histograms, sub-totals, cross tabulations, roll-up/drill down, data cubes
- Expensive task (computationally)

Motivation

- Perform data mining and online analytical processing (OLAP) in a efficient way
- OLAP is :
  - Resource-demanding
  - Calls for parallelization
- OLAP operations:
  - Pivot
  - Roll-up
  - Cube

Related work

- Ng. et al develop a collection of parallel algorithms to data cube construction in low cost PCs (Clustering)
- PARSIMONY: provides a parallel and scalable infrastructure for multidimensional analyses
- There are commercial solutions like Oracle and IBM that also implement their parallel algorithms
- This paper propose a new direction: OLAP and data mining should rely on Linear Algebra

Cross tabulation

- Provides a **summary** of a data extracted from raw source
- Example:

Model	Year	Color	Sales
Chevy	1990	Red	5
Chevy	1990	Blue	87
Ford	1990	Green	64
Ford	1990	Blue	99
Ford	1991	Red	8
Ford	1991	Blue	7

- How many vehicles sold per colour and model?

Cross tabulation

- How many vehicles sold per colour and model?
- Selected Color and Model as attributes and Sales as a measure
- Answer is:

Sum of Sales	Model		
Color	Chevy	Ford	Grand Total
Blue	87	106	193
Green		64	64
Red	5	8	13
Grand Total	92	178	270

In this paper: solve this problem with Linear Algebra.
But how we can parallelize?

OLAP - Cube

- Cross tabulation summaries:
  - Computationally expensive
  - Long time (large datasets)
- OLAP cube compute all dimensions
- Calculate all possible options
- Summarize the table
  - Works like a cache of values
  - Easy to compute and access data in time

			Sales
Chevy	1990	*Blue*	87
Chevy	1990	*Red*	5
Ford	1990	*Blue*	99
Ford	1990	*Green*	64
Ford	1991	*Blue*	7
Ford	1991	*Red*	8
Chevy	1990	*ALL*	92
Ford	1990	*ALL*	163
Ford	1991	*ALL*	15
Chevy	*ALL*	*Blue*	87
Chevy	*ALL*	*Red*	5
Ford	*ALL*	*Blue*	106
Ford	*ALL*	*Green*	64
Ford	*ALL*	*Red*	8
ALL	1990	*Blue*	186
ALL	1990	*Green*	64
ALL	1990	*Red*	5
ALL	1991	*Blue*	7
ALL	1991	*Red*	8
Chevy	*ALL*	*ALL*	92
Ford	*ALL*	*ALL*	178
ALL	1990	*ALL*	255
ALL	1991	*ALL*	15
ALL	*ALL*	*Blue*	193
ALL	*ALL*	*Green*	64
ALL	*ALL*	*Red*	13
ALL	*ALL*	*ALL*	270

Cross tabulation – Linear Algebra

- Three matrices:
  - Two associated with dimensions (attributes) – A and B
  - Measure or Metric
- Divide-and-conquer principle, with matrix multiplication:

$$[\,R|S\,] \cdot \left[\frac{U}{V}\right] = R \cdot U + S \cdot V$$

- OLAP cross-tabulation can be expressed by: $t_A \cdot [\![T]\!]_M \cdot t_B^\circ$
- A, B is dimensions and M is the measure

$$t_A(x,r) = \begin{cases} 1 & \text{if } T(A,r) = x \\ 0 & \text{otherwise} \end{cases}$$

Cross tabulation – Linear Algebra

	1	2	3	4	5	6
Chevy	1	1	0	0	0	0
Ford	0	0	1	1	1	1

$$|Model| \xleftarrow{t_{Model}} n$$

	1	2	3	4	5	6
Blue	0	1	0	1	0	1
Green	0	0	1	0	0	0
Red	1	0	0	0	1	0

$$|Color| \xleftarrow{t_{Color}} n$$

$$t_{Color} \cdot t^{\circ}_{Model} =$$

	Chevy	*Ford*
Blue	1	2
Green	0	1
Red	1	1

Cross tabulation – Linear Algebra

$$t_{Color} \cdot [\![T]\!]_{Sales} \cdot t^{\circ}_{Model} =$$

	Chevy	*Ford*
Blue	87	106
Green	0	64
Red	5	8

$$ctab_{Color,Model;Sales}(T) =$$

	Chevy	*Ford*	ALL
Blue	87	106	193
Green	0	64	64
Red	5	8	13
ALL	92	178	270

Rolling-up on functional dependences

- Rolling-up means replacing a dimension by another which is more general in some sense (eg. grouping, classification, containment).

Model	Year	Color	Sales	Month	Season
Chevy	1990	Red	5	March	Spring
Chevy	1990	Blue	87	April	Spring
Ford	1990	Green	64	August	Summer
Ford	1990	Blue	99	October	Autumn
Ford	1991	Red	8	January	Winter
Ford	1991	Blue	7	January	Winter

- Also works for checking functional dependences

Rolling-up on functional dependences

- Rolling-up means replacing a dimension by another which is more general in some sense (eg. grouping, classification, containment).

$$t_{Season} \cdot t^{\circ}_{Month} =$$

	January	*March*	*April*	*August*	*October*
Spring	0	1	1	0	0
Summer	0	0	0	1	0
Autumn	0	0	0	0	1
Winter	2	0	0	0	0

- Also works for checking functional dependences

Rolling-up on functional dependences

- Rolling-up means replacing a dimension by another which is more general in some sense (eg. grouping, classification, containment).

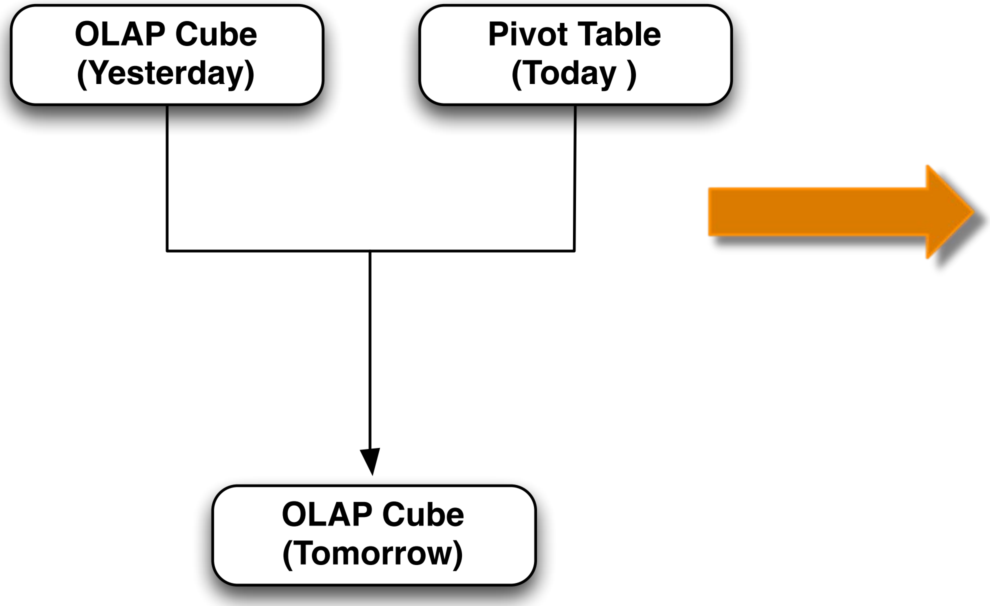
$$(t_{Season \leftarrow Month} \oplus id) \cdot ctab_{Month,Model;Sales}(T) =$$

	Chevy	*Ford*	ALL
Spring	92	0	92
Summer	0	64	64
Autumn	0	99	99
Winter	0	15	15
ALL	92	178	270

- Also works for checking functional dependences

Incremental construction

- Cross tabulations defined by Linear Algebra is amenable to incremental constructions



$$t''_A = \left[\, t_A \middle| t'_A \,\right]$$
$$t''_B = \left[\, t_B \middle| t'_B \,\right]$$
$$[\![T; T']\!]_M = [\![T]\!]_M \oplus [\![T']\!]_M$$

- Advantage: is not necessary to build all the CUBE every single day!

Higher dimensionality - OLAP

- Consider n-dimensions: aggregate, group-by, cross tabulations and cube
- Generalization based on Khatri-Rao product
  - Works like a Cartesian product
- Khatri-Rao product:

$$s = \begin{bmatrix} 5 & 87 & 64 & 99 & 8 & 7 \end{bmatrix}$$

$$6 \xleftarrow{s \odot id} 6 \; = \begin{bmatrix} 5 & 0 & 0 & 0 & 0 & 0 \\ 0 & 87 & 0 & 0 & 0 & 0 \\ 0 & 0 & 64 & 0 & 0 & 0 \\ 0 & 0 & 0 & 99 & 0 & 0 \\ 0 & 0 & 0 & 0 & 8 & 0 \\ 0 & 0 & 0 & 0 & 0 & 7 \end{bmatrix}$$

Higher-dimensional OLAP

- All dimensions
- Whole dimension part
- Raw-data table
- The Khatri-Roa of:
    - tModel and tColor

	1	2	3	4	5	6
Chevy	1	1	0	0	0	0
Ford	0	0	1	1	1	1

$$|Model| \xleftarrow{t_{Model}} n$$

	1	2	3	4	5	6
Blue	0	1	0	1	0	1
Green	0	0	1	0	0	0
Red	1	0	0	0	1	0

$$|Color| \xleftarrow{t_{Color}} n$$

		1	2	3	4	5	6
Chevy	*Blue*	0	1	0	0	0	0
Chevy	*Green*	0	0	0	0	0	0
Chevy	*Red*	1	0	0	0	0	0
Ford	*Blue*	0	0	0	1	0	1
Ford	*Green*	0	0	1	0	0	0
Ford	*Red*	0	0	0	0	1	0

Higher-dimensional OLAP

- All dimensions
- Whole dimension part
- Raw-data table

$$t_{Model \times Year \times Color}$$
$$= t_{Model} \odot t_{Year} \odot t_{Color}$$
$$=$$

			1	2	3	4	5	6
Chevy	1990	*Blue*	0	1	0	0	0	0
Chevy	1990	*Green*	0	0	0	0	0	0
Chevy	1990	*Red*	1	0	0	0	0	0
Chevy	1991	*Blue*	0	0	0	0	0	0
Chevy	1991	*Green*	0	0	0	0	0	0
Chevy	1991	*Red*	0	0	0	0	0	0
Ford	1990	*Blue*	0	0	0	1	0	0
Ford	1990	*Green*	0	0	1	0	0	0
Ford	1990	*Red*	0	0	0	0	0	0
Ford	1991	*Blue*	0	0	0	0	0	1
Ford	1991	*Green*	0	0	0	0	0	0
Ford	1991	*Red*	0	0	0	0	1	0

Higher-dimensional OLAP

- All dimensions
- Whole dimension part
- Raw-data table

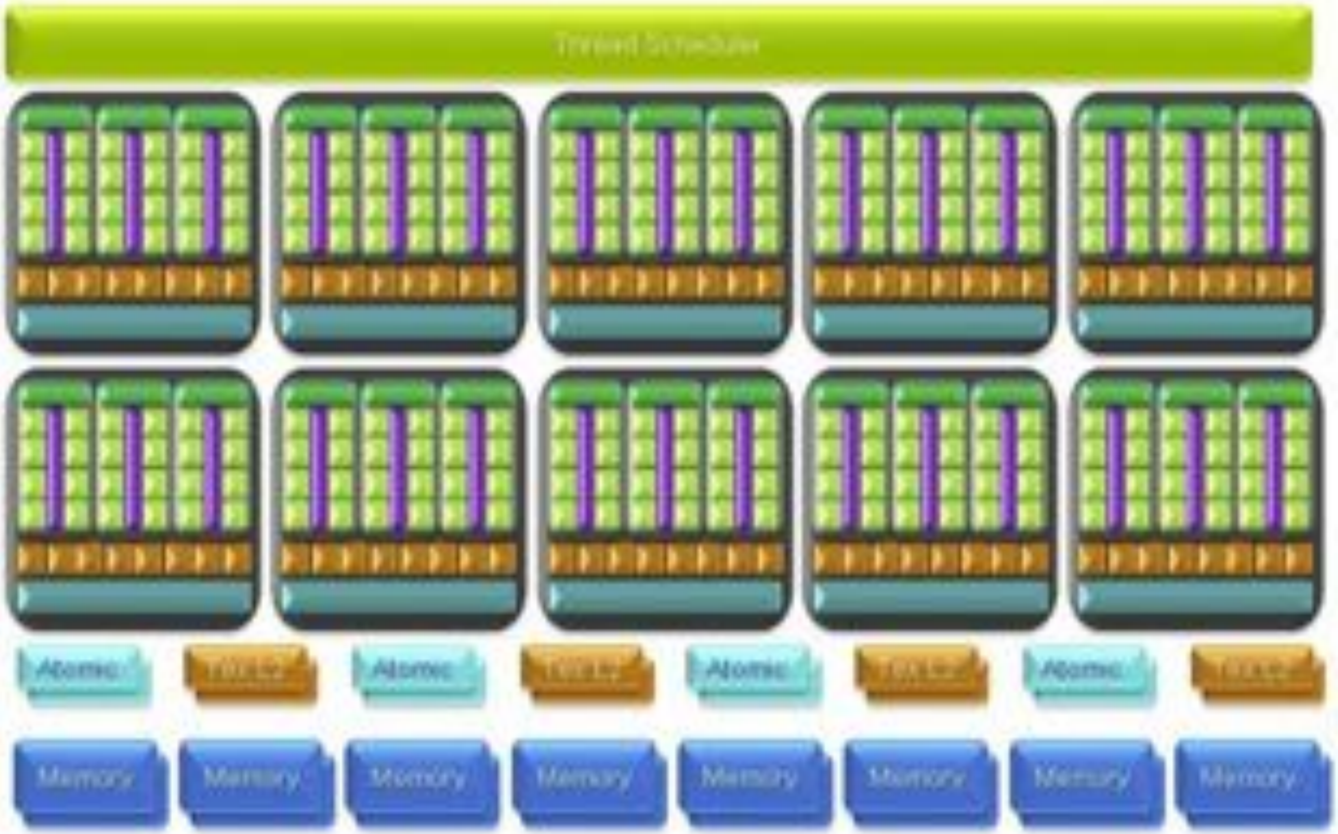
$$t_{Model \times Year \times Color} \cdot [\![T]\!]_{Sales} \cdot !^{\circ} =$$

			Sales
Chevy	1990	*Blue*	87
Chevy	1990	*Green*	0
Chevy	1990	*Red*	5
Chevy	1991	*Blue*	0
Chevy	1991	*Green*	0
Chevy	1991	*Red*	0
Ford	1990	*Blue*	99
Ford	1990	*Green*	64
Ford	1990	*Red*	0
Ford	1991	*Blue*	7
Ford	1991	*Green*	0
Ford	1991	*Red*	8

Conclusion and future work

- OLAP computationally problematic
- Parallelization is already possible, but not with linear algebra
- Encoding OLAP in concepts of Linear Algebra – formal method
- Rely on theory of parallel sparse matrix/matrix multiplication to achieve parallelism
- Cross tabulation is incremental
- Future:
  - Extending LA for other OLAP features
  - Implement in Multi-core and GPU and replace the OpenOffice/ LibreOffice pivot table calculator

Future work (GPGPU)



Questions?