

# How to generate large-scale data from small-scale real-world data sets?

Gang Lu

Institute of Computing Technology,  
Chinese Academy of Sciences

BigDataBench Tutorial  
MICRO 2014 Cambridge, UK



中国科学院计算所  
INSTITUTE OF COMPUTING TECHNOLOGY

# Motivation

- Benchmarking big data systems
  - The first thing is to obtain **BIG** data
- Obtaining **REAL** big data sets?
  - Large companies possess a lot of data
    - Confidentiality issue (User privacy)
  - Transferring big data sets is rather expensive

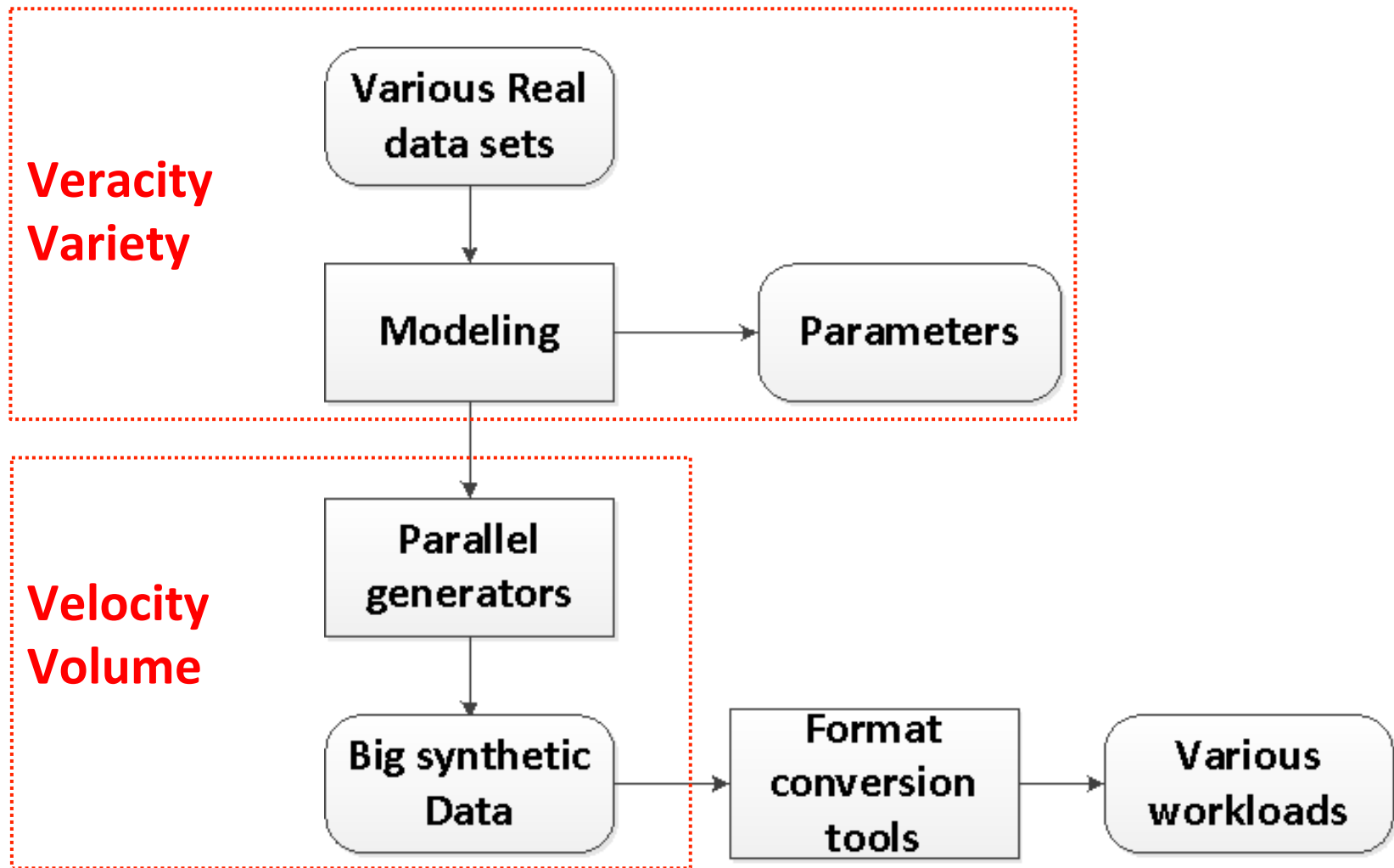
Is it possible to use large scale synthetic data?

# Goals

- Generating synthetic data to satisfy 4V properties of big data
  - Volume
  - Velocity
  - Variety
  - Veracity

**Big Data Generator Suite(BDGS)**

# Architecture of BDGS



# Veracity and Variety

- From real world data, we can get:

Veracity	Variety		
Data set	Data type	Data source	Application domains
Wikipedia Entries	un-structured	text data	Search engine
Amazon Movie Reviews	semi-structure	text data	E-commence
Google Web Graph	un-structured	graph data	Search engine
Facebook Social Graph	un-structured	graph data	Social network
E-commence Transaction	structured	table data	E-commence
Profsearch Person Resume	semi-structured	table data	Search engine

# Original size of real data sets

data sets	data size
Wikipedia Entries	4,300,000 English articles
Amazon Movie Reviews	7,911,684 reviews
Google Web Graph	875713 nodes, 5105039 edges
Facebook Social Network	4039 nodes, 88234 edges
E-commence Transaction	table1: 4 columns, 38658 rows. table2: 6 columns, 242735 rows
Person Resumes Data	278956 resumes

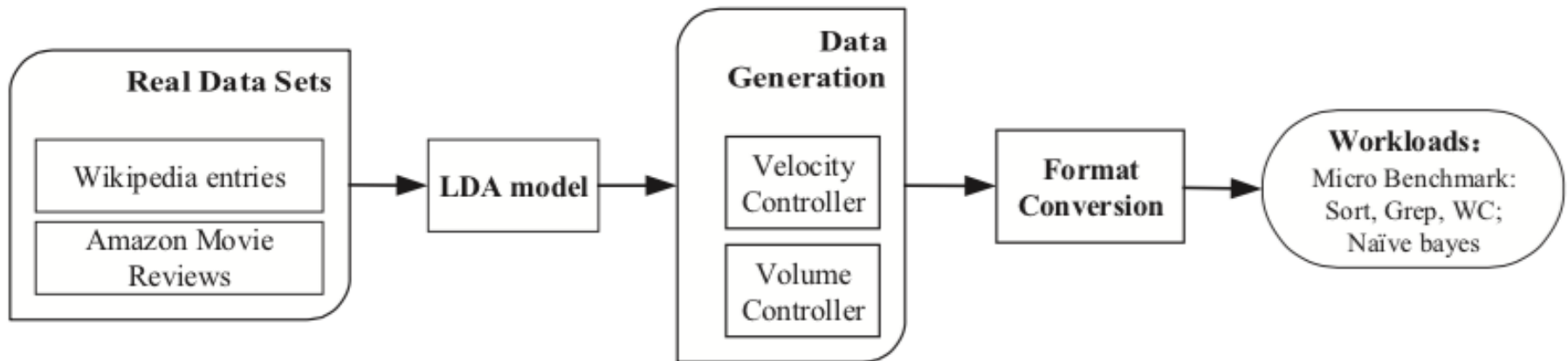
**Use BDGS to scale up these data sets**

# What does BDGS provide?

- Text generator
- Graph generator
- Table generator

# Text generator

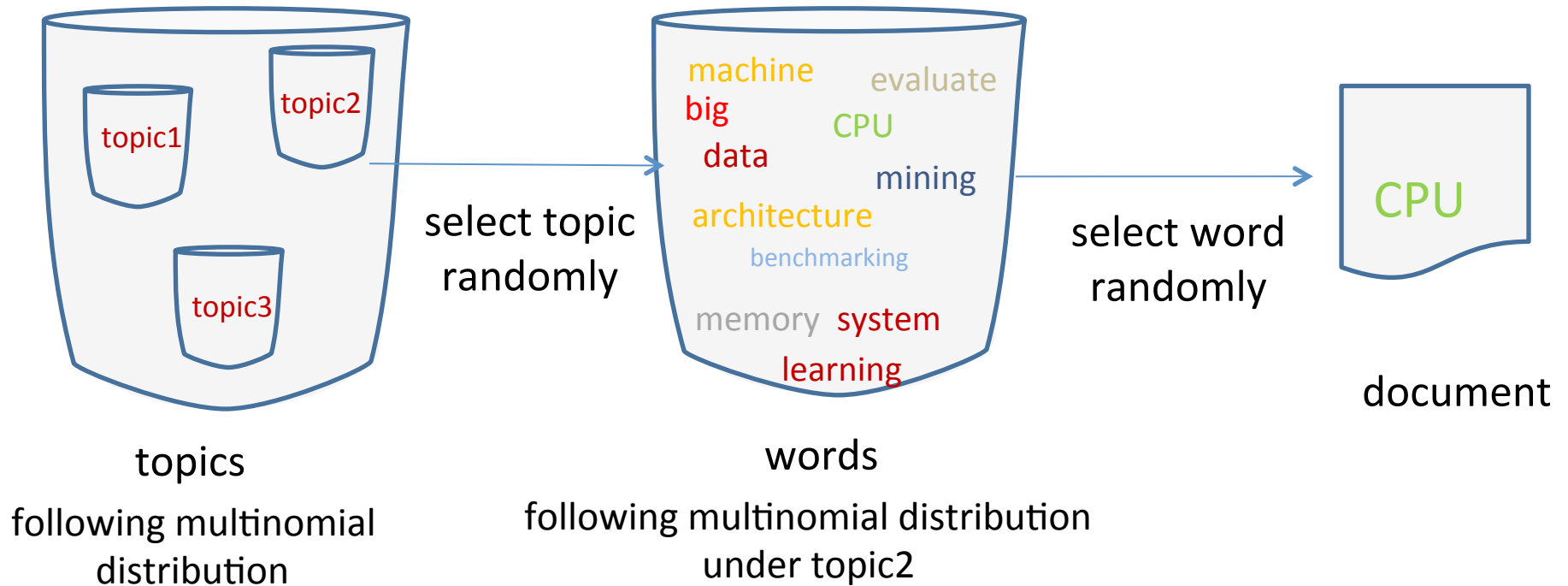
- Use LDA (Latent Dirichlet Allocation) (David M Blei, et al.) to generate text corpus.
  - **Topic model**
    - To model the imformaion of semantic level
  - Widely used in machine learning and natural language processing





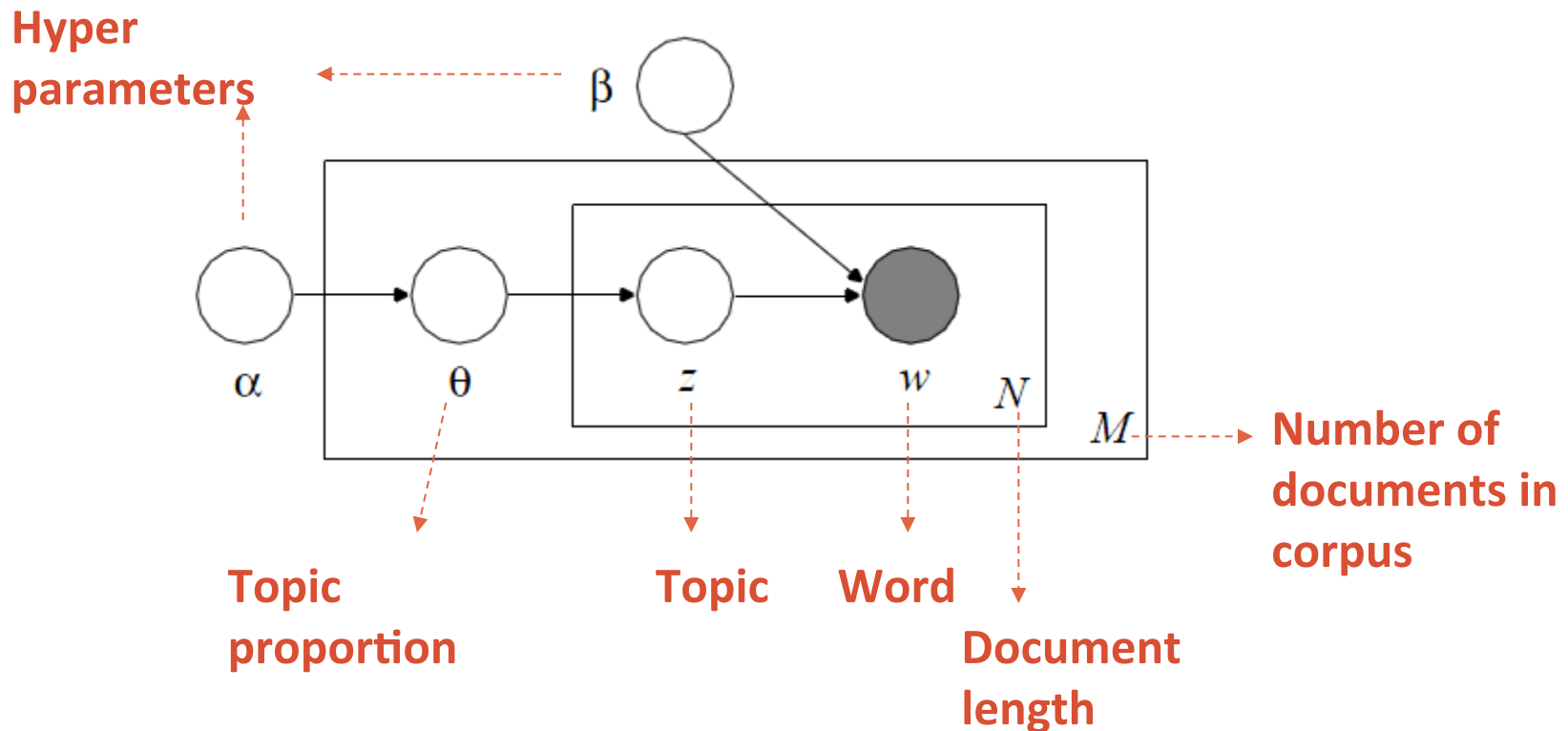
# Text generator

## ■ How to generate a new document



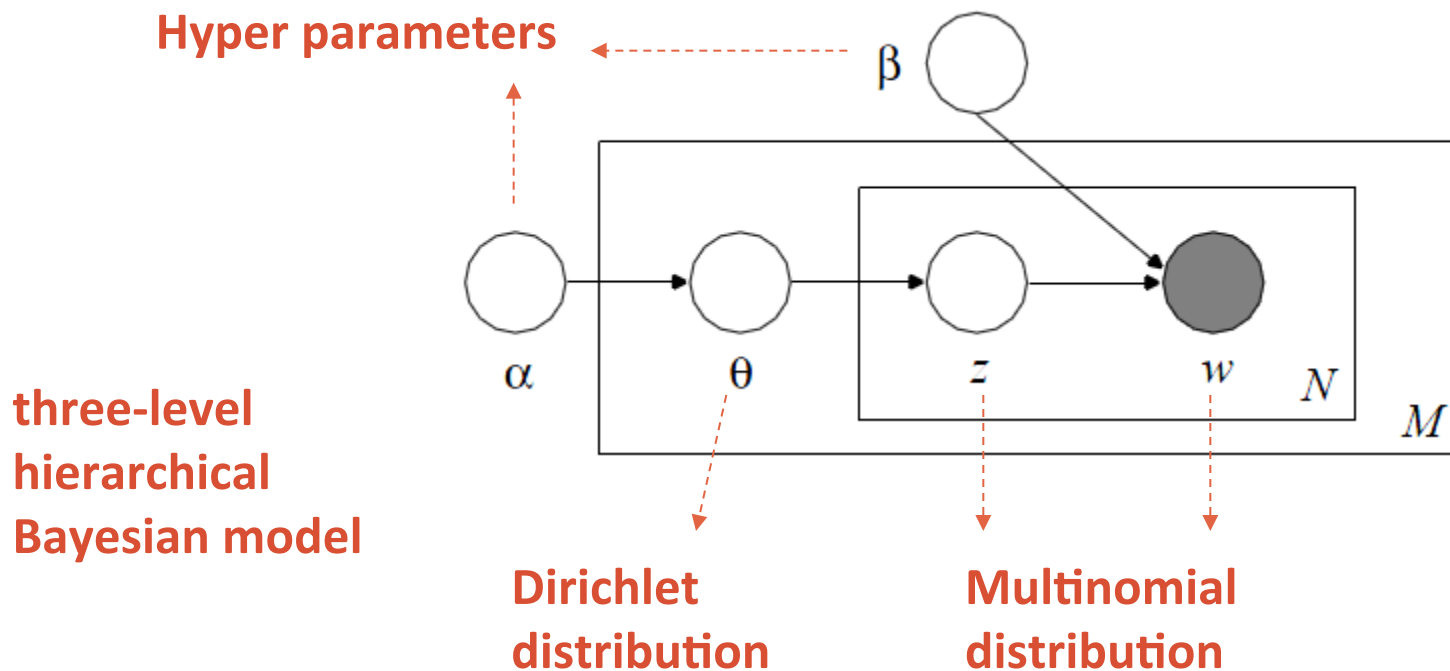
Progress of generating a new document

# Latent dirichlet allocation



David M Blei, et al., "Latent dirichlet allocation," the Journal of machine Learning research, vol. 3, pp. 993–1022, 2003.

# Latent dirichlet allocation



$$p(\theta, \mathbf{z}, \mathbf{w} | \alpha, \beta) = p(\theta | \alpha) \prod_{n=1}^N p(z_n | \theta) p(w_n | z_n, \beta)$$

David M Blei, et al., "Latent dirichlet allocation," *the Journal of machine Learning research*, vol. 3, pp. 993–1022, 2003.

## Topics

gene	0.04
dna	0.02
genetic	0.01
...	

life	0.02
evolve	0.01
organism	0.01
...	

brain	0.04
neuron	0.02
nerve	0.01
...	

data	0.02
number	0.02
computer	0.01
...	

## Documents

### Seeking Life's Bare (Genetic) Necessities

COLD SPRING HARBOR, NEW YORK—How many genes does an organism need to survive? Last week at the genome meeting here,\* two genome researchers with radically different approaches presented complementary views of the basic genes needed for life. One research team, using computer analyses to compare known genomes, concluded that today's organisms can be sustained with just 250 genes, and that the earliest life forms required a mere 128 genes. The other researcher mapped genes in a simple parasite and estimated that for this organism, 800 genes are plenty to do the job—but that anything short of 100 wouldn't be enough.

Although the numbers don't match precisely, those predictions

\* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 8 to 12.

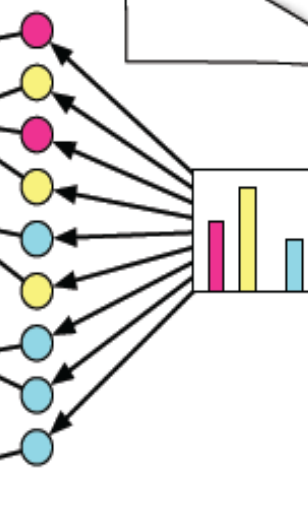
"are not all that far apart," especially in comparison to the 75,000 genes in the human genome, notes Siv Andersson, a University in Sweden, who arrived at the 800 number. But coming up with a consensus answer may be more than just a genetic numbers game, particularly as more and more genomes are completely mapped and sequenced. "It may be a way of organizing any newly sequenced genome," explains Arcady Mushegian, a computational molecular biologist at the National Center for Biotechnology Information (NCBI) in Bethesda, Maryland. Comparing an



**Stripping down.** Computer analysis yields an estimate of the minimum modern and ancient genomes.

SCIENCE • VOL. 272 • 24 MAY 1996

## Topic proportions and assignments



We can use expectation-maximization algorithm to determine  $\alpha$  and  $\beta$

# How to use it to generate texts?

- Go into the directory of *BigDataGeneratorSuite*

```
gen_text_data.sh  
<model name>  
<number of files>  
<number of lines>  
<number of words>  
<output dir >
```

Parameters	Explannation
<b>model name</b>	the name of model used to generate new data (lda_wiki1w or amazonMR)
<b>No. of files</b>	the number of files to be generated
<b>No. of lines</b>	number of lines in each file
<b>No. of words</b>	number of words in each line
<b>Output dir</b>	output director

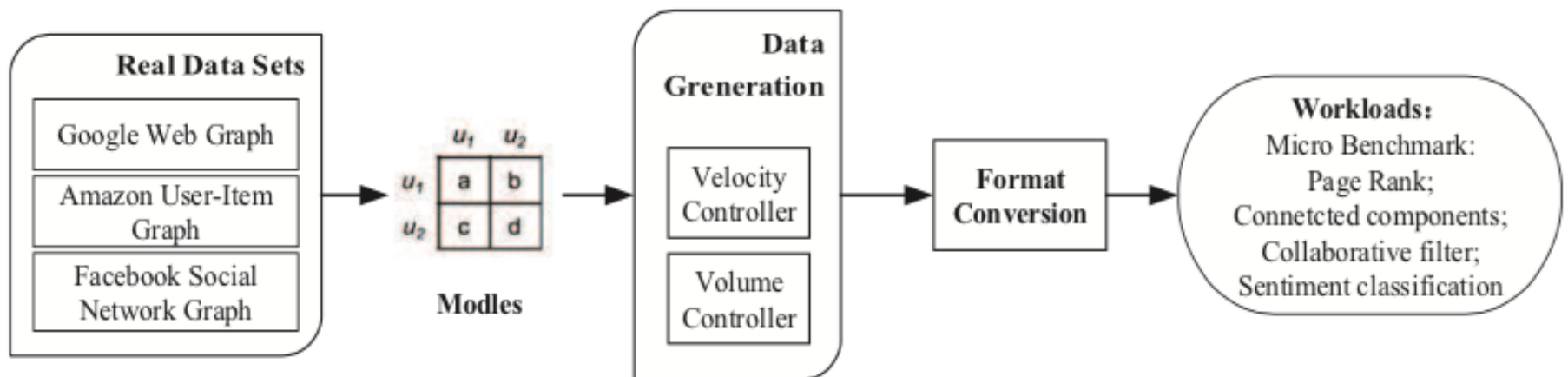
- An example

- `sh gen_text_data.sh lda_wiki1w 10 100 1000 gen_data/`

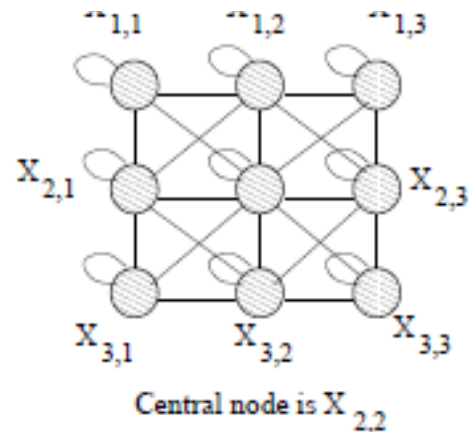
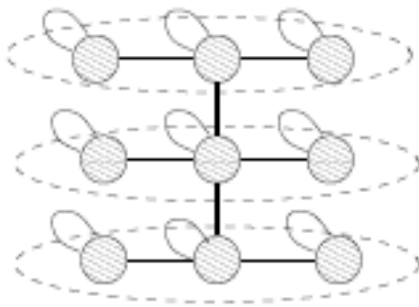
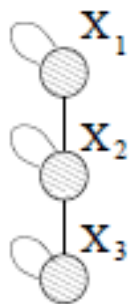
Note: Installation of the GSL-GNU Scientific Library is needed.

# Graph generator

- Use the Stochastic Kronecker Graph model (Jure Leskovec, et al.) to generate graph
  - Used also by graph 500
  - Different from Graph 500, our graph is application specific, the stochastic kronecker initiator is obtained from real representative data set of



# Deterministic Kronecker Graph



1: has edge

0: no edge

(a) Graph  $K_1$

(b) Intermediate stage

(c) Graph  $K_2 = K_1 \otimes K_1$

1	1	0
1	1	1
0	1	1

self similar

$K_1$	$K_1$	0
$K_1$	$K_1$	$K_1$
0	$K_1$	$K_1$

(d) Adjacency matrix

(e) Adjacency matrix

Jure Leskovec, et al., "Kronecker graphs: An approach to modeling networks," *The Journal of Machine Learning Research*, vol. 11, pp. 985–1042, 2010.

# Stochastic Kronecker Graph

The probability with which the cell generate a edge

	$u_1$	$u_2$
$u_1$	a	b
$u_2$	c	d

(a)  $2 \times 2$  Stochastic Kronecker initiator  $\mathcal{P}_1$

	$v_1$	$v_2$	$v_3$	$v_4$
$v_1$	a·a	a·b	b·a	b·b
$v_2$	a·c	a·d	b·c	b·d
$v_3$	c·a	c·b	d·a	d·b
$v_4$	c·c	c·d	d·c	d·d

(b) Probability matrix  
 $\mathcal{P}_2 = \mathcal{P}_1 \otimes \mathcal{P}_1$

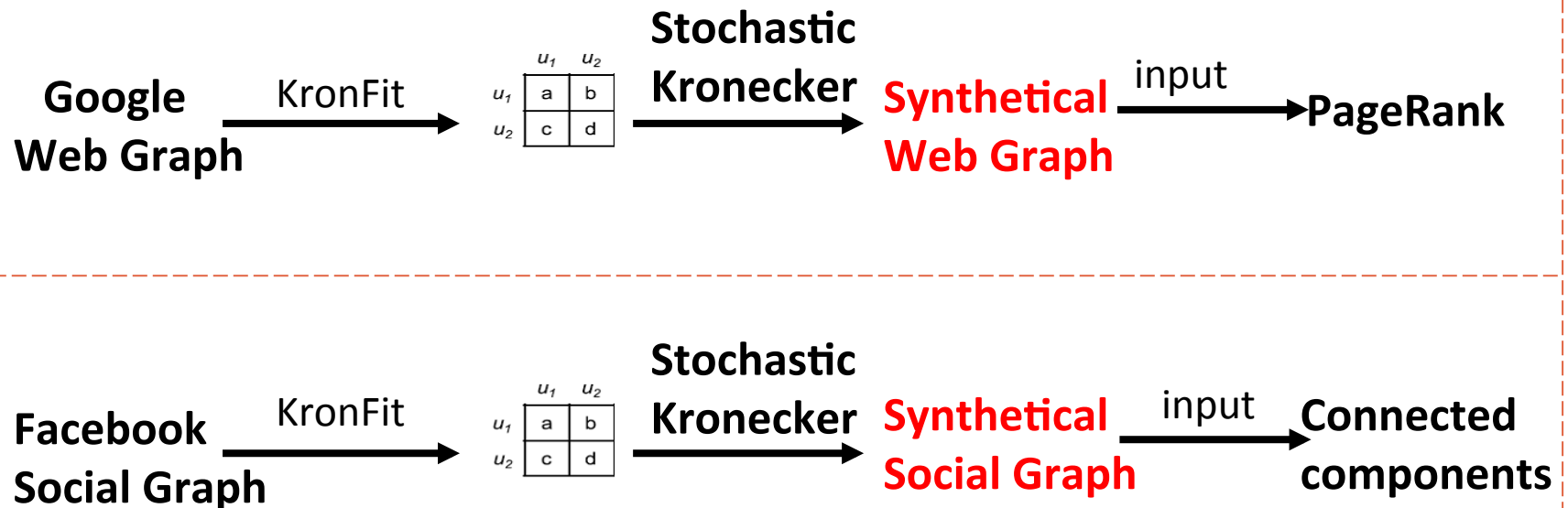
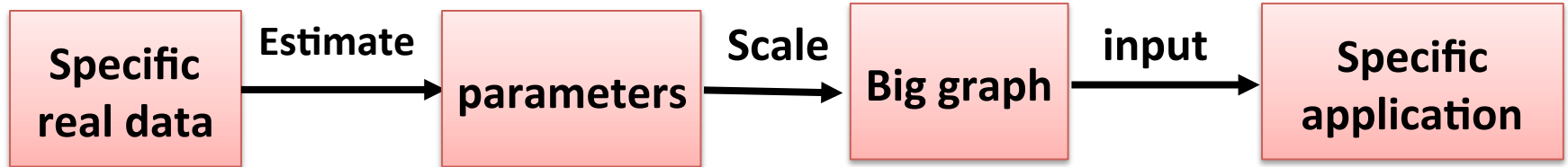
	$v_1$	$v_2$	$v_3$	$v_4$
$v_1$	a	b	a	b
$v_2$	<b>a</b>	d	<b>c</b>	d
$v_3$	a	b	a	b
$v_4$	<b>c</b>	d	<b>d</b>	d

(c) Alternative view of  $\mathcal{P}_2 = \mathcal{P}_1 \otimes \mathcal{P}_1$

Jure Leskovec, et al., "Kronecker graphs: An approach to modeling networks," *The Journal of Machine Learning Research*, vol. 11, pp. 985–1042, 2010.



# Application-specific



# How to use it to generate graphs?

- Go into the directory of *BigDataGeneratorSuite*

```
gen_kronecker_graph  
<output file>  
<matrix>  
<iteration>  
<random seed>
```

Parameters	Explanation
output file	output file name (default: 'graph.txt')
matrix	Matrix (in Matlab notation) (default: '0.9 0.5; 0.5 0.1')
iteration	Iteration of using kronecker product (default: 5)
random seed	time seed of random algorithm (default: 0)

- An example

- `sh gen kronecker graph -o:../data-outfile/amazon gen.txt -m:"0.7196 0.6313; 0.4833 0.3601" -i:23`

# Table generator

## ■ Related structured table

### ■ Parallel Data Generation Framework (Tilman Rabl, et al.)

- PDGF is also used by BigBench and TPC-DS
- using XML configuration files for data description and distribution

## ■ Semi-structured resumes

- choose a mix of fields, each field follows bernoulli distribution

# How to use it to generate tables?

- Go into the directory of *BigDataGeneratorSuite*

```
pdgf.jar  
-l schema.xml  
-l generation.xml  
-sf 2000
```

Parameters	Explanation
schema.xml	the schema configuration: the structure of the data and the generation rules
generation.xml	the generation configuration defines the output and post-processing of generated data
sf	A multiple increase in the reference data base 100 000

- An example

- `java -XX:NewRatio=1 -jar pdgf.jar -l demo-schema.xml -l demo-generation.xml -c -s -sf 2000`



**Any**  
**Questions**