15-826: Multimedia Databases and Data Mining

Lecture #19: Tensor decompositions

C. Faloutsos

Problem

- Q: who-calls-whom-when – patterns?
  - Triplets (source-ip, dest-ip, port#)
  - KB (subject, verb, object)
Conclusions

• Q: who-calls-whom-when – patterns?
  – Triplets (source-ip, dest-ip, port#)
  – KB (subject, verb, object)

• A: Tensor analysis (PARAFAC)
  – http://www.tensortoolbox.org/

\[
\mathbf{X} = \lambda_1 \mathbf{a}_1 \mathbf{b}_1 \mathbf{c}_1 + \ldots + \lambda_N \mathbf{a}_R \mathbf{b}_R \mathbf{c}_R
\]

Must-read Material

• [Graph-Textbook] Ch.16.

• Tensors survey: Papalexakis, Faloutsos, Sidiropoulos Tensor for Data Mining and Data Fusion: Models, Applications, and Scalable Algorithms ACM Trans. on Intelligent Systems and Technology, 8,2, Oct. 2016. (local copy)
Outline

Goal: ‘Find similar / interesting things’
- Intro to DB
- Indexing - similarity search
- Data Mining

Indexing - Detailed outline
- primary key indexing
- secondary key / multi-key indexing
- spatial access methods
- fractals
- text
- Singular Value Decomposition (SVD)
  - …
- Tensors
- multimedia
- …
Outline

• Motivation - Definitions
• Tensor tools
• Case studies

Most of foils by

• Dr. Tamara Kolda (Sandia N.L.)
  • cmr.ca.sandia.gov/~tgkolda

• Prof. Jimeng Sun (GaTech)
  • www.cc.gatech.edu/people/jimeng-sun

3h tutorial: www.cs.cmu.edu/~christos/TALKS/SDM-tut-07/
Motivation 1: Why “matrix”?

- Why matrices are important?

Examples of Matrices: Graph - social network

|       | John | Peter | Mary | Nick | ...
|-------|------|-------|------|------|-------
| John  | 0    | 11    | 22   | 55   | ...
| Peter | 5    | 0     | 6    | 7    | ...
| Mary  |      |       |      |      | ...
| Nick  |      |       |      |      | ...
| ...   |      |       |      |      | ...
### Examples of Matrices:

**cloud of n-d points**

<table>
<thead>
<tr>
<th>chol#</th>
<th>blood#</th>
<th>age</th>
</tr>
</thead>
</table>
| John  | 13     | 11  | 22 | 55 |...
| Peter | 5      | 4   | 6  | 7  |...
| Mary  | ...    | ... | ...| ...|...
| Nick  | ...    | ... | ...| ...|...
| ...   | ...    | ... | ...| ...|...

### Examples of Matrices:

**Market basket**

- **market basket** as in Association Rules

<table>
<thead>
<tr>
<th>milk</th>
<th>bread</th>
<th>choc.</th>
<th>wine</th>
</tr>
</thead>
</table>
| John | 13    | 11    | 22   | 55  |...
| Peter| 5     | 4     | 6    | 7   |...
| Mary | ...   | ...   | ...  | ... |...
| Nick | ...   | ...   | ...  | ... |...
| ...  | ...   | ...   | ...  | ... |...
### Examples of Matrices: Documents and terms

<table>
<thead>
<tr>
<th>Paper#1</th>
<th>Paper#2</th>
<th>Paper#3</th>
<th>Paper#4</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>mining</td>
<td>classif.</td>
<td>tree</td>
<td>...</td>
</tr>
<tr>
<td>13</td>
<td>11</td>
<td>22</td>
<td>55</td>
<td>...</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>6</td>
<td>7</td>
<td>...</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

### Examples of Matrices: Authors and terms

<table>
<thead>
<tr>
<th>John</th>
<th>Peter</th>
<th>Mary</th>
<th>Nick</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>mining</td>
<td>classif.</td>
<td>tree</td>
<td>...</td>
</tr>
<tr>
<td>13</td>
<td>11</td>
<td>22</td>
<td>55</td>
<td>...</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>6</td>
<td>7</td>
<td>...</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Examples of Matrices:
sensor-ids and time-ticks

<table>
<thead>
<tr>
<th></th>
<th>temp1</th>
<th>temp2</th>
<th>humid.</th>
<th>pressure</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>t1</td>
<td>13</td>
<td>11</td>
<td>22</td>
<td>55</td>
<td>...</td>
</tr>
<tr>
<td>t2</td>
<td>5</td>
<td>4</td>
<td>6</td>
<td>7</td>
<td>...</td>
</tr>
<tr>
<td>t3</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>t4</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Motivation: Why tensors?

- Q: what is a tensor?
Motivation 2: Why tensor?

- A: N-D generalization of matrix:

<table>
<thead>
<tr>
<th>KDD' 17</th>
<th>data</th>
<th>mining</th>
<th>classif.</th>
<th>tree</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>13</td>
<td>11</td>
<td>22</td>
<td>55</td>
<td>...</td>
</tr>
<tr>
<td>Peter</td>
<td>5</td>
<td>4</td>
<td>6</td>
<td>7</td>
<td>...</td>
</tr>
<tr>
<td>Mary</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>Nick</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Copyright (c) 2019 C. Faloutsos
## Tensors are useful for 3 or more modes

Terminology: ‘mode’ (or ‘aspect’):

<table>
<thead>
<tr>
<th></th>
<th>data</th>
<th>mining</th>
<th>classif.</th>
<th>tree</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mode#3</td>
<td>13</td>
<td>11</td>
<td>22</td>
<td>55</td>
<td>...</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>4</td>
<td>6</td>
<td>7</td>
<td>...</td>
</tr>
<tr>
<td></td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td></td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>Mode#2</td>
<td>15-826</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Mode (== aspect) #1

## Motivating Applications

- Why matrices are important?
- Why tensors are useful?
  - P1: social networks
  - P2: web mining
P1: Social network analysis

- Traditionally, people focus on static networks and find community structures.
- We plan to monitor the change of the community structure over time.

P2: Web graph mining

- How to order the importance of web pages?
  - Kleinberg’s algorithm HITS
  - PageRank
  - Tensor extension on HITS (TOPHITS)
    • context-sensitive hypergraph analysis
Outline

• Motivation – Definitions
• Tensor tools
• Case studies

Tensor Basics

Copyright (c) 2019 C. Faloutsos
Answer to both: tensor factorization

• Recall: (SVD) matrix factorization: finds blocks

\[ \mathbf{M} \]

\[ \mathbf{U} \]

\[ \mathbf{V} \]

\[ \mathbf{W} \]

‘meat-eaters’ ‘vegetarians’ ‘kids’

‘steaks’ ‘plants’ ‘cookies’

Answer to both: tensor factorization

• PARAFAC decomposition

\[ \text{subject} \]

\[ \text{verb} \]

\[ \text{object} \]

\[ \text{politicians} \]

\[ \text{artists} \]

\[ \text{athletes} \]
Answer: tensor factorization

- PARAFAC decomposition
- Results for who-calls-whom-when
  - 4M x 15 days

\[ \text{caller} = \sum \lambda_I \mathbf{a}_I \otimes b_T \otimes c_T \]

Goal: extension to >=3 modes

\[ \mathbf{X} = \sum \lambda_r \mathbf{a}_r \otimes \mathbf{b}_r \otimes \mathbf{c}_r \]

Example of outer product ‘o’:
Goal: extension to >=3 modes

\[ X \approx [\lambda; A, B, C] = \sum_{r} \lambda_r \, a_r \circ b_r \circ c_r \]

Suppose

\[ R=1 \]
\[ a_1=(1,2,3,4) \]
\[ b_1=(2,2,2) \]
\[ c_1=(10,11) \]
\[ \lambda_1=7 \]

Copyright (c) 2019 C. Faloutsos
Goal: extension to >=3 modes

\[ X \approx \llbracket \lambda ; A, B, C \rrbracket = \sum_{r} \lambda_r \ a_r \circ b_r \circ c_r \]

Suppose

- \( r=1 \)
- \( a_1 = (1,2,3,4) \)
- \( b_1 = (2,2,2) \)
- \( c_1 = (10,11) \)
- \( \lambda_1 = 7 \)

\[ X(1,1,1) = 7 \times 1 \times 2 \times 10 \]
\[ X(3,1,2) = 7 \times 3 \times 2 \times 11 \]
\[ X(5,1,1) = N/A - TRICK QUESTION \]
Goal: extension to >=3 modes

\[ \mathbf{X} \approx [\lambda; \mathbf{A}, \mathbf{B}, \mathbf{C}] = \sum \lambda_r \mathbf{a}_r \odot \mathbf{b}_r \odot \mathbf{c}_r \]

Suppose
- \( r=1 \)
- \( \mathbf{a}_1=(1,2,3,4) \)
- \( \mathbf{b}_1=(2,2,2) \)
- \( \mathbf{c}_1=(10,11) \)
- \( \lambda_1=7 \)

\( \mathbf{X}(1,1,1)=7*1*2*10 \)
\( \mathbf{X}(3,1,2)= 7*3*2*11 \)
\( \mathbf{X}(5,1,1)= N/A - TRICK QUESTION \)

Main points:

- 2 major types of tensor decompositions: PARAFAC and Tucker
- both can be solved with `alternating least squares' `(ALS)
- Details follow
Specially Structured Tensors

- Tucker Tensor
  \[ \mathbf{X} = \mathbf{g} \times_1 \mathbf{U} \times_2 \mathbf{V} \times_3 \mathbf{W} \]
  \[ = \sum_r \sum_s \sum_t g_{rst} \mathbf{u}_r \circ \mathbf{v}_s \circ \mathbf{w}_t \]
  \[ \equiv [\mathbf{g}; \mathbf{U}, \mathbf{V}, \mathbf{W}] \]

- Kruskal Tensor
  \[ \mathbf{X} = \sum_r \lambda_r \mathbf{u}_r \circ \mathbf{v}_r \circ \mathbf{w}_r \]
  \[ \equiv [\lambda; \mathbf{U}, \mathbf{V}, \mathbf{W}] \]

Our Notation
Specially Structured Tensors

- **Tucker Tensor**
  \[ \mathcal{X} = G_{1} U \otimes V \otimes W \]
  \[ = \sum_{r} \sum_{s} \sum_{t} g_{rst} u_{r} \circ v_{s} \circ w_{t} \]
  \[ \equiv [G; U, V, W] \]

In matrix form:
\[ X_{(1)} = U G_{(1)} (W \otimes V)^{T} \]
\[ X_{(2)} = V G_{(2)} (W \otimes U)^{T} \]
\[ X_{(3)} = W G_{(3)} (V \otimes U)^{T} \]

\[ \text{vec}(\mathcal{X}) = (W \otimes V \otimes U) \text{vec}(G) \]

- **Kruskal Tensor**
  \[ \mathcal{X} = \sum_{r} \lambda_{r} u_{r} \circ v_{r} \circ w_{r} \]
  \[ \equiv [\lambda; U, V, W] \]

In matrix form:
\[ X_{(1)} = U \Lambda (W \otimes V)^{T} \]
\[ X_{(2)} = V \Lambda (W \otimes U)^{T} \]
\[ X_{(3)} = W \Lambda (V \otimes U)^{T} \]

\[ \text{vec}(\mathcal{X}) = (W \otimes V \otimes U) \lambda \]

Tensor Decompositions
Tucker Decomposition - intuition

- author x keyword x conference
- A: author x author-group
- B: keyword x keyword-group
- C: conf. x conf-group
- $G$: how groups relate to each other

Needs elaboration!

Intuition behind core tensor

- 2-d case: co-clustering
- [Dhillon et al. Information-Theoretic Co-clustering, KDD’03]
\[
\begin{align*}
\begin{bmatrix}
5 & 0 & 0 \\
5 & 0 & 0 \\
0 & 5 & 0 \\
0 & 5 & 0 \\
0 & 0 & 5 \\
0 & 0 & 5 \\
\end{bmatrix}
\begin{bmatrix}
3 & 0 & 0 \\
0 & 3 & 0 \\
2 & 2 & 0 \\
\end{bmatrix}
\begin{bmatrix}
36 & 0.36 & 0.28 & 0 & 0 & 0 \\
0 & 0 & 0 & 36 & 0.36 & 0.28 \\
\end{bmatrix}
\end{align*}
\]

\[m = n \cdot l \cdot k\]
Tucker Decomposition

- Proposed by Tucker (1966)
- AKA: Three-mode factor analysis, three-mode PCA, orthogonal array decomposition
- $A$, $B$, and $C$ generally assumed to be orthonormal (generally assume they have full column rank)
- $G$ is not diagonal
- Not unique

Given $A$, $B$, $C$, the optimal core is:

$$G = [X; A^\top, B^\top, C^\top]$$

Recall the equations for converting a tensor to a matrix:

$$X_{(1)} = AG_{(1)}(C \otimes B)^\top$$
$$X_{(2)} = BG_{(2)}(C \otimes A)^\top$$
$$X_{(3)} = CG_{(3)}(B \otimes A)^\top$$

$$\text{vec}(X) = (C \otimes B \otimes A)\text{vec}(G)$$
Kronecker product

\[ A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, \quad B = \begin{bmatrix} 10 & 20 & 30 \end{bmatrix} \]

\[ m1 \times n1 \quad m2 \times n2 \]

\[ A \otimes B = \begin{bmatrix} 1 \ast B & 2 \ast B \\ 3 \ast B & 4 \ast B \end{bmatrix} \]

\[ = \begin{bmatrix} 1 \ast 10 & 1 \ast 20 & 1 \ast 30 & 2 \ast 10 & 2 \ast 20 & 2 \ast 30 \\ 3 \ast 10 & 3 \ast 20 & 3 \ast 30 & 4 \ast 10 & 4 \ast 20 & 4 \ast 30 \end{bmatrix} \]

Outline

- Motivation – Definitions
- Tensor tools
- Case studies
  - Tensor Basics
  - Tucker
  - PARAFAC
CANDECOMP/Parafac Decomposition

\[ \mathbf{X} \approx \sum_{r=1}^{R} \lambda_r \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r \]

- CANDECOMP = Canonical Decomposition (Carroll & Chang, 1970)
- PARAFAC = Parallel Factors (Harshman, 1970)
- Core is diagonal (specified by the vector \( \lambda \))
- Columns of \( \mathbf{A}, \mathbf{B}, \text{ and } \mathbf{C} \) are not orthonormal
- If \( R \) is minimal, then \( R \) is called the rank of the tensor (Kruskal, 1977)
- Can have rank( \( \mathbf{X} \) ) \( > \) min{\( I, J, K \)}

Tucker vs. PARAFAC Decompositions

- **Tucker**
  - Variable transformation in each mode
  - Core \( \mathbf{G} \) may be dense
  - \( \mathbf{A}, \mathbf{B}, \text{ and } \mathbf{C} \) generally orthonormal
  - Not unique

- **PARAFAC**
  - Sum of rank-1 components
  - No core, i.e., superdiagonal core
  - \( \mathbf{A}, \mathbf{B}, \text{ and } \mathbf{C} \) may have linearly dependent columns
  - Generally unique
Tensor tools - summary

- Two main tools
  - PARAFAC
  - Tucker
- Both find row-, column-, tube-groups
  - but in PARAFAC the three groups are identical
- To solve: Alternating Least Squares

- Toolbox: from Tamara Kolda:

Outline

- Motivation - Definitions
- Tensor tools
- Case studies
  - P1: web graph mining (‘TOPHITS’)
  - P2: phone-call patterns
  - P3: N.E.L.L. (never ending language learner)
  - P4: network traffic
  - P5: FaceBook activity
P1: Web graph mining

• How to order the importance of web pages?
  – Kleinberg’s algorithm HITS
  – PageRank

P1: Web graph mining

Kleinberg’s Hubs and Authorities (the HITS method)

Sparse adjacency matrix and its SVD:

\[ x_{ij} = \begin{cases} 1 & \text{if page } i \text{ links to page } j \\ 0 & \text{otherwise} \end{cases} \]

\[ X \approx \sum \sigma_r h_r \circ a_r \]

authority scores for 1st topic

hub scores for 1st topic

authority scores for 2nd topic

hub scores for 2nd topic

HITS Authorities on Sample Data

We started our crawl from http://www-neos.mcs.anl.gov/neos, and crawled 4700 pages, resulting in 560 cross-linked hosts.
Three-Dimensional View of the Web

Observe that this tensor is very sparse!

Topical HITS (TOPHITS)

Main Idea: Extend the idea behind the HITS model to incorporate term (i.e., topical) information.

$$X \approx \sum_{r=1}^{R} \lambda_r \mathbf{h}_r \odot \mathbf{a}_r$$
Topical HITS (TOPHITS)

Main Idea: Extend the idea behind the HITS model to incorporate term (i.e., topical) information.

\[ \mathbf{X} \approx \sum_{r=1}^{R} \lambda_r \mathbf{h}_r \circ \mathbf{a}_r \circ \mathbf{t}_r \]

TOPHITS Terms & Authorities on Sample Data

Tensor PARAFAC

Copyright (c) 2019 C. Faloutsos
Outline

• Motivation - Definitions
• Tensor tools
• Case studies
  – P1: web graph mining (‘TOPHITS’)
  – P2: phone-call patterns
  – P3: N.E.L.L. (never ending language learner)
  – P4: network traffic
  – P5: FaceBook activity

P2: Anomaly detection in time-evolving graphs

• Anomalous communities in phone call data:
  – European country, 4M clients, data over 2 weeks

~200 calls to EACH receiver on EACH day!
P2: Anomaly detection in time-evolving graphs

- Anomalous communities in phone call data:
  - European country, 4M clients, data over 2 weeks

1 caller

5 receivers

4 days of activity

~200 calls to EACH receiver on EACH day!
P2: Anomaly detection in time-evolving graphs

- Anomalous communities in phone call data:
  - European country, 4M clients, data over 2 weeks


Outline

- Motivation - Definitions
- Tensor tools
- Case studies
  - P1: web graph mining (‘TOPHITS’)
  - P2: phone-call patterns
  - P3: N.E.L.L. (never ending language learner)
  - P4: network traffic
  - P5: FaceBook activity
GigaTensor: Scaling Tensor Analysis Up By 100 Times – Algorithms and Discoveries

U Kang Evangelos Abhay Christos Papalexakis Harpale Faloutsos

KDD 2012

P3: N.E.L.L. analysis

- NELL: Never Ending Language Learner
  - Q1: dominant concepts / topics?
  - Q2: synonyms for a given new phrase?

“Eric Clapton plays guitar”
“Barrack Obama is the president of U.S.”

NELL (Never Ending Language Learner)
Nonzeros = 144M

subjects (26M) verbs (48M)
objects (26M)
A1: Concept Discovery

- Concept Discovery in Knowledge Base

<table>
<thead>
<tr>
<th>Concept 1: &quot;Web Protocol&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>internet protocol</td>
</tr>
<tr>
<td>file software suite</td>
</tr>
<tr>
<td>data</td>
</tr>
<tr>
<td>np1 'stream' np2</td>
</tr>
<tr>
<td>np1 'marketing' np2</td>
</tr>
<tr>
<td>np1 'dating' np2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Concept 2: &quot;Credit Cards&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>credit information</td>
</tr>
<tr>
<td>debt</td>
</tr>
<tr>
<td>library number</td>
</tr>
<tr>
<td>np1 'card' np2</td>
</tr>
<tr>
<td>np1 'report' np2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Concept 3: &quot;Health System&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>health provider</td>
</tr>
<tr>
<td>child providers</td>
</tr>
<tr>
<td>home system</td>
</tr>
<tr>
<td>np1 'care' np2</td>
</tr>
<tr>
<td>np1 'insurance' np2</td>
</tr>
<tr>
<td>np1 'service' np2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Concept 4: &quot;Family Life&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>life rest</td>
</tr>
<tr>
<td>family part</td>
</tr>
<tr>
<td>body years</td>
</tr>
<tr>
<td>np2 'of' my np1</td>
</tr>
<tr>
<td>np2 'of' his np1</td>
</tr>
<tr>
<td>np2 'of' her np1</td>
</tr>
</tbody>
</table>
A2: Synonym Discovery

- Synonym Discovery in Knowledge Base

<table>
<thead>
<tr>
<th>(Given) Noun Phrase</th>
<th>(Discovered) Potential Synonyms</th>
</tr>
</thead>
<tbody>
<tr>
<td>pollutants</td>
<td>dioxin, sulfur dioxide, greenhouse gases, particulates, nitrogen oxide, air pollutants, cholesterol</td>
</tr>
<tr>
<td>disabilities</td>
<td>infections, dizziness, injuries, diseases, drowsiness, stiffness, injuries</td>
</tr>
<tr>
<td>vodafone</td>
<td>verizon, comcast</td>
</tr>
<tr>
<td>Christian history</td>
<td>European history, American history, Islamic history, history</td>
</tr>
<tr>
<td>disbelief</td>
<td>dismay, disgust, astonishment</td>
</tr>
<tr>
<td>cyberpunk</td>
<td>online-gaming</td>
</tr>
<tr>
<td>soul</td>
<td>body</td>
</tr>
</tbody>
</table>
Outline

• Motivation - Definitions
• Tensor tools
• Case studies
  – P1: web graph mining (‘TOPHITS’)
  – P2: phone-call patterns
  – P3: N.E.L.L. (never ending language learner)
  – P4: network traffic
  – P5: FaceBook activity

ParCube: Sparse Parallelizable Tensor Decompositions

Evangelos E. Papalexakis, Christos Faloutsos, Nikos Sidiropoulos, ECML/PKDD 2012

Evangelos E. Papalexakis
Email: epapalex@cs.ucr.edu
Web: http://www.cs.ucr.edu/~epapalex
Tensor applications such as content-based network analysis and visualization. The list continues, including authors use tensors for social network analysis on the algorithm HITS, incorporating textual-topical information. In [7] and [6] deemed representative: In [3], the authors extend the well-known link analysis technique, tensor applications in data mining is long, however, we single out a few that we report on Table 2.

Potential Synonyms

• Modes: src IP, dst IP, port #
• ~ Port Scanning Attack

P4: LBNL Network Data

- modes: src IP, dst IP, port #
- ~ port scanning attack

P5: Facebook Wall posts

- modes: wall-owner, poster, timestamp
- discovery: birthday-like event.
Conclusions

• Q: who-calls-whom-when – patterns?
  – Triplets (source-ip, dest-ip, port#)
  – KB (subject, verb, object)
• A: Tensor analysis (PARAFAC)
  – http://www.tensortoolbox.org/

\[ \mathbf{X} = \sum_{r=1}^{R} \lambda_r \mathbf{C}_r \mathbf{D}_r \mathbf{E}_r \]

References

• Inderjit S. Dhillon, Subramanyam Mallela, Dharmendra S. Modha: Information-theoretic co-clustering. KDD 2003: 89-98
• Jimeng Sun, Spiros Papadimitriou, Philip Yu. Window-based Tensor Analysis on High-dimensional and Multi-aspect Streams, Proc. of the Int. Conf. on Data Mining (ICDM), Hong Kong, China, Dec 2006