Mining Latent Entity Structures

Jiawei Han and Chi Wang
Computer Science, University of Illinois at Urbana-Champaign
June 25, 2014
Outline

1. Introduction to mining latent entity structures
2. Mining latent topic hierarchies
3. Mining latent entity relations
4. Mining latent entity concepts
5. Trends and research problems
Motivation of Mining Latent Entity Structures

- The prevalence of unstructured data
- Structures are useful for knowledge discovery

Too expensive to be structured by human: Automated & scalable

Up to 85% of all information is unstructured -- estimated by industry analysts

Vast majority of the CEOs expressed frustration over their organization’s inability to glean insights from available data -- IBM study with 1500+ CEOs
Information Overload: A Critical Problem in Big Data Era

By 2020, information will double every 73 days

-- G. Starkweather (Microsoft), 1992

Unstructured or loosely structured data are prevalent
Example: Research Publications

Every year, hundreds of thousands papers are published

- Unstructured data: paper text
- Loosely structured entities: authors, venues
Example: News Articles

Every day, >90,000 news articles are produced

- Unstructured data: news content
- Loosely structured entities: persons, locations, organizations, ...
Example: Social Media

Every second, >150K tweets are sent out
- Unstructured data: tweet content
- Loosely structured entities: twitters, hashtags, URLs, ...

Darth Vader
@darthvader · May 4
I’m the reason for the season, Happy May the Fourth be with you
#maythefourthbewithyou

The White House
Happy Star Wars Day! I'm not building a Death Star.
#maythefourthbewithyou

URL
flic.kr/p/75XWNY
wh.gov/Pttj

URL
wh.gov/Pttj
A Semi-Structured Framework (or Model) for Unstructured and Loosely-Structured Data
Useful Latent Structures: Topics, Concepts, Relations

- Top 10 active politicians regarding healthcare issues?
- Influential high-tech companies in Silicon Valley?
- Top 10 researchers in data mining and their specializations?
- Academic family on support vector machine?
Mining Latent Structures: Topics, Concepts, Relations

- Concepts
  - Discover the type of entities and disambiguate their mentions

- Unstructured text & linked entities -> structured hierarchies

- Relations
  - Discover latent relations between entities

Concepts, Topics, Relations
What Power Can We Gain if More Structures Can Be Discovered?

- Structured database queries
- Information network analysis, ...

<table>
<thead>
<tr>
<th>Christos Faloutsos</th>
</tr>
</thead>
<tbody>
<tr>
<td>Carnegie Mellon University, 2000 – 2010</td>
</tr>
<tr>
<td>University of Maryland, 1986 – 1997</td>
</tr>
<tr>
<td><strong>Author Rankings</strong></td>
</tr>
<tr>
<td>Temporal and Spatial Databases: 190</td>
</tr>
<tr>
<td>Database and Information System: 9</td>
</tr>
<tr>
<td>Data Mining: 4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Frequent co-authors</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spiros Papadimitriou</td>
</tr>
<tr>
<td>Jimeng Sun</td>
</tr>
<tr>
<td>Agma J. M. Traina</td>
</tr>
<tr>
<td>Hanghang Tong</td>
</tr>
<tr>
<td>Jia-Yu Pan</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Advisees</th>
<th>Period</th>
<th>Prob</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hanghang Tong</td>
<td>2006 – 2010</td>
<td>1.00</td>
</tr>
<tr>
<td>Mukund Seshadri</td>
<td>2008 – 2010</td>
<td>1.00</td>
</tr>
<tr>
<td>B. Aditya Prakash</td>
<td>2009 – 2010</td>
<td>1.00</td>
</tr>
<tr>
<td>Leman Akoglu</td>
<td>2008 – 2010</td>
<td>1.00</td>
</tr>
<tr>
<td>U. Kang</td>
<td>2009 – 2010</td>
<td>1.00</td>
</tr>
<tr>
<td>Fan Guo</td>
<td>2008 – 2009</td>
<td>1.00</td>
</tr>
<tr>
<td>Jure Leskovec</td>
<td>2005 – 2008</td>
<td>1.00</td>
</tr>
<tr>
<td>Dacheng Tao</td>
<td>2006 – 2008</td>
<td>1.00</td>
</tr>
</tbody>
</table>
Structures Facilitate Multi-Dimensional Analysis: An EventCube Experiment

Democrats ask: can **health care bill** be saved?

Deem this: a frantic push for a health-reform bill that congress has yet to see
Distribution along Multiple Dimensions
Query ‘health care bill’ in news data
Entity Analysis and Profiling

Topic distribution for “Stanford University”
Analyzing, Mining, and Exploring a Topical Hierarchy System

AMETHYST [DANILEVSKY ET AL. 13]
Structures Facilitate Heterogeneous Information Network Analysis

Real-world data: Multiple object types and/or multiple link types
What Can Be Mined in Structured Information Networks

Example: DBLP: A Computer Science bibliographic database

<table>
<thead>
<tr>
<th>Knowledge hidden in DBLP Network</th>
<th>Mining Functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Who are the leading researchers on Web search?</td>
<td>Ranking</td>
</tr>
<tr>
<td>Who are the peer researchers of Jure Leskovec?</td>
<td>Similarity Search</td>
</tr>
<tr>
<td>Whom will Christos Faloutsos collaborate with?</td>
<td>Relationship Prediction</td>
</tr>
<tr>
<td>Which types of relationships are most influential for an author to decide her topics?</td>
<td>Relation Strength Learning</td>
</tr>
<tr>
<td>How was the field of Data Mining emerged or evolving?</td>
<td>Network Evolution</td>
</tr>
<tr>
<td>Which authors are rather different from his/her peers in IR?</td>
<td>Outlier/anomaly detection</td>
</tr>
</tbody>
</table>
Similarity Search: Find Similar Objects in Networks Guided by Meta-Paths

Who are very similar to Christos Faloutsos?

Meta-Path: **Meta-level description** of a path between two objects

**Schema of the DBLP Network**

**Different meta-paths lead to very different results!**

<table>
<thead>
<tr>
<th>Rank</th>
<th>Author</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Christos Faloutsos</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>Spiros Papadimitriou</td>
<td>0.127</td>
</tr>
<tr>
<td>3</td>
<td>Jimeng Sun</td>
<td>0.12</td>
</tr>
<tr>
<td>4</td>
<td>Jia-Yu Pan</td>
<td>0.114</td>
</tr>
<tr>
<td>5</td>
<td>Agma J. M. Traina</td>
<td>0.110</td>
</tr>
<tr>
<td>6</td>
<td>Jure Leskovec</td>
<td>0.096</td>
</tr>
<tr>
<td>7</td>
<td>Caetano Traina Jr.</td>
<td>0.096</td>
</tr>
<tr>
<td>8</td>
<td>Hanghang Tong</td>
<td>0.091</td>
</tr>
<tr>
<td>9</td>
<td>Deepayan Chakrabarti</td>
<td>0.083</td>
</tr>
<tr>
<td>10</td>
<td>Flip Korn</td>
<td>0.053</td>
</tr>
</tbody>
</table>

**Christos’s students or close collaborators**

**Author-Paper-Author (APA)**

<table>
<thead>
<tr>
<th>Rank</th>
<th>Author</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Christos Faloutsos</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>Jiawei Han</td>
<td>0.842</td>
</tr>
<tr>
<td>3</td>
<td>Rakesh Agrawal</td>
<td>0.838</td>
</tr>
<tr>
<td>4</td>
<td>Jian Pei</td>
<td>0.8</td>
</tr>
<tr>
<td>5</td>
<td>Charu C. Aggarwal</td>
<td>0.739</td>
</tr>
<tr>
<td>6</td>
<td>H. V. Jagadish</td>
<td>0.705</td>
</tr>
<tr>
<td>7</td>
<td>Raghu Ramakrishnan</td>
<td>0.697</td>
</tr>
<tr>
<td>8</td>
<td>Nick Koudas</td>
<td>0.689</td>
</tr>
<tr>
<td>9</td>
<td>Surajit Chaudhuri</td>
<td>0.677</td>
</tr>
<tr>
<td>10</td>
<td>Divesh Srivastava</td>
<td>0.661</td>
</tr>
</tbody>
</table>

**Author-Paper-Venue-Paper-Author (APVPA)**

**Similar reputation at similar venues**
Similarity Search: PathSim Measure Helps Find Peer Objects in Long Tails

Anhai Doan
- CS, Wisconsin
- Database area
- PhD: 2002

PathSim
[Sun et al. 11]

Meta-Path: Author-Paper-Venue-Paper-Author (APVPA)

<table>
<thead>
<tr>
<th>Rank</th>
<th>P-PageRank</th>
<th>SimRank</th>
<th>PathSim</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>AnHai Doan</td>
<td>AnHai Doan</td>
<td>AnHai Doan</td>
</tr>
<tr>
<td>2</td>
<td>Philip S. Yu</td>
<td>Douglas W. Cornell</td>
<td>Jignesh M. Patel</td>
</tr>
<tr>
<td>3</td>
<td>Jiawei Han</td>
<td>Adam Silberstein</td>
<td>Amol Deshpande</td>
</tr>
<tr>
<td>4</td>
<td>Hector Garcia-Molina</td>
<td>Samuel DeFazio</td>
<td>Jun Yang</td>
</tr>
<tr>
<td>5</td>
<td>Gerhard Weikum</td>
<td>Curt Ellmann</td>
<td>Renée J. Miller</td>
</tr>
</tbody>
</table>

- Jignesh Patel
  - CS, Wisconsin
  - Database area
  - PhD: 1998

- Amol Deshpande
  - CS, Maryland
  - Database area
  - PhD: 2004

- Jun Yang
  - CS, Duke
  - Database area
  - PhD: 2001
PathPredict: Meta-Path Based Relationship Prediction

- Meta path-guided prediction of links and relationships

- Insight: Meta path relationships among similar typed links share similar semantics and are comparable and inferable

- Bibliographic network: Co-author prediction (A—P—A)
Meta-Path Based Co-authorship Prediction

- Co-authorship prediction: Whether two authors start to collaborate
- Co-authorship encoded in meta-path: Author-Paper-Author
- Topological features encoded in meta-paths

<table>
<thead>
<tr>
<th>Meta-Path</th>
<th>Semantic Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A \rightarrow P \rightarrow P \rightarrow A$</td>
<td>$a_i$ cites $a_j$</td>
</tr>
<tr>
<td>$A \leftarrow P \rightarrow P \rightarrow A$</td>
<td>$a_i$ is cited by $a_j$</td>
</tr>
<tr>
<td>$A \rightarrow P \rightarrow V \rightarrow P \rightarrow A$</td>
<td>$a_i$ and $a_j$ publish in the same venues</td>
</tr>
<tr>
<td>$A \rightarrow P \rightarrow A \rightarrow P \rightarrow A$</td>
<td>$a_i$ and $a_j$ are co-authors of the same authors</td>
</tr>
<tr>
<td>$A \rightarrow P \rightarrow T \rightarrow P \rightarrow A$</td>
<td>$a_i$ and $a_j$ write the same topics</td>
</tr>
<tr>
<td>$A \rightarrow P \leftarrow P \rightarrow P \rightarrow A$</td>
<td>$a_i$ cites papers that cite $a_j$</td>
</tr>
<tr>
<td>$A \leftarrow P \leftarrow P \rightarrow P \rightarrow A$</td>
<td>$a_i$ is cited by papers that are cited by $a_j$</td>
</tr>
<tr>
<td>$A \rightarrow P \leftarrow P \rightarrow P \rightarrow A$</td>
<td>$a_i$ and $a_j$ cite the same papers</td>
</tr>
<tr>
<td>$A \leftarrow P \leftarrow P \rightarrow P \rightarrow A$</td>
<td>$a_i$ and $a_j$ are cited by the same papers</td>
</tr>
</tbody>
</table>

The prediction power of each meta-path derived by logistic regression:

<table>
<thead>
<tr>
<th>Meta Path</th>
<th>$p$-value</th>
<th>significance level$^1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A \rightarrow P \rightarrow P \rightarrow A$</td>
<td>0.0378</td>
<td>**</td>
</tr>
<tr>
<td>$A \rightarrow P \leftarrow P \rightarrow A$</td>
<td>0.0077</td>
<td>***</td>
</tr>
<tr>
<td>$A \rightarrow P \rightarrow V \rightarrow P \rightarrow A$</td>
<td>1.2974e-174</td>
<td>*****</td>
</tr>
<tr>
<td>$A \rightarrow P \rightarrow A \rightarrow P \rightarrow A$</td>
<td>1.1484e-126</td>
<td>*****</td>
</tr>
<tr>
<td>$A \rightarrow P \rightarrow T \rightarrow P \rightarrow A$</td>
<td>3.4867e-51</td>
<td>*****</td>
</tr>
<tr>
<td>$A \rightarrow P \rightarrow P \rightarrow P \rightarrow A$</td>
<td>0.7459</td>
<td></td>
</tr>
<tr>
<td>$A \rightarrow P \leftarrow P \leftarrow P \rightarrow A$</td>
<td>0.0647</td>
<td>*</td>
</tr>
<tr>
<td>$A \rightarrow P \rightarrow P \leftarrow P \rightarrow A$</td>
<td>9.7641e-11</td>
<td>****</td>
</tr>
<tr>
<td>$A \rightarrow P \leftarrow P \rightarrow P \leftarrow A$</td>
<td>0.0966</td>
<td>*</td>
</tr>
</tbody>
</table>

$^1$ *: $p < 0.1$; **: $p < 0.05$; ***: $p < 0.01$; ****: $p < 0.001$
Heterogeneous Network Helps Personalized Recommendation

- Users and items with limited feedback are connected by a variety of paths.
- Different users may require different models: Relationship heterogeneity makes personalized recommendation models easier to define.

Collaborative filtering methods suffer from the data sparsity issue.

A small set of users & items have a large number of ratings.

Most users and items have a small number of ratings.

Personalized recommendation with heterogeneous networks [Yu et al. 14a]
Personalized Recommendation in Heterogeneous Networks

- Datasets:
- Methods to compare:
  - Popularity: Recommend the most popular items to users
  - Co-click: Conditional probabilities between items
  - NMF: Non-negative matrix factorization on user feedback
  - Hybrid-SVM: Use Rank-SVM to utilize both user feedback and information network

<table>
<thead>
<tr>
<th>Name</th>
<th>#Items</th>
<th>#Users</th>
<th>#Ratings</th>
<th>#Entities</th>
<th>#Links</th>
</tr>
</thead>
<tbody>
<tr>
<td>IM100K</td>
<td>943</td>
<td>1360</td>
<td>89,626</td>
<td>60,905</td>
<td>146,013</td>
</tr>
<tr>
<td>Yelp</td>
<td>11,537</td>
<td>43,873</td>
<td>229,907</td>
<td>285,317</td>
<td>570,634</td>
</tr>
</tbody>
</table>

**Winner:** HeteRec personalized recommendation (HeteRec-p)
Outline

1. Introduction to mining latent entity structures
2. Mining latent topic hierarchies
3. Mining latent entity relations
4. Mining latent entity concepts
5. Trends and research problems
Mining Latent Topical Hierarchy

Topics

Unstructured text & linked entities
-> structured hierarchies

entity

text
Topic Hierarchy: Summarize the Data with Multiple Granularity

- Top 10 researchers in data mining?
  - And their specializations?

- Important research areas in SIGIR conference?
Methodologies of Topic Mining

A. Traditional bag-of-words topic modeling

B. Extension of topic modeling
   i) Flat -> hierarchical
   ii) Unigrams -> ngrams
   iii) Text -> text + entity

C. An integrated framework: CATHY
   i) Recursive topic discovery
   ii) Phrase mining
   iii) Phrase and entity ranking
Methodologies of Topic Mining

A. Traditional bag-of-words topic modeling

B. Extension of topic modeling
   i) Flat -> hierarchical
   ii) Unigrams -> ngrams
   iii) Text -> text + entity

C. An integrated framework: CATHY
   i) Recursive topic discovery
   ii) Phrase mining
   iii) Phrase and entity ranking
A. Bag-of-Words Topic Modeling

- Widely studied technique for text analysis
  - Summarize themes/aspects
  - Facilitate navigation/browsing
  - Retrieve documents
  - Segment documents
  - Many other text mining tasks

- Represent each document as a bag of words: all the words within a document are exchangeable

- Probabilistic approach
Topic:
Multinomial Distribution over Words

A document is modeled as a sample of mixed topics

[ Criticism of government response to the hurricane primarily consisted of criticism of its response to the approach of the storm and its aftermath, specifically in the delayed response to the flooding of New Orleans. ... 80% of the 1.3 million residents of the greater New Orleans metropolitan area evacuated] ...[ Over seventy countries pledged monetary donations or other assistance]. ...

How can we discover these topic word distributions from a corpus?
Routine of Generative Models

- Model design: assume the documents are generated by a certain process

- Model Inference: Fit the model with observed documents to recover the unknown parameters $\Theta$

Two representative models: pLSA and LDA

Criticism of government response to the hurricane...
Probabilistic Latent Semantic Analysis (PLSA) [Hofmann 99]

- $k$ topics: $k$ multinominal distributions over words
- $D$ documents: $D$ multinominal distributions over topics

Generative process: we will generate each token in each document $d$ according to $\phi, \theta$
PLSA – Model Design

- **$k$ topics**: $k$ multinomial distributions over words
- **$D$ documents**: $D$ multinomial distributions over topics

To generate a token in document $d$:
1. Sample a topic label $z$ according to $\theta_d$ (e.g. $z=1$)
2. Sample a word $w$ according to $\phi_z$ (e.g. $w=$government)
PLSA – Model Inference

What parameters are most likely to generate the observed corpus?

To generate a token in document $d$:
1. Sample a topic label $z$ according to $\theta_d$ (.4 .3 .3) (e.g. $z=1$)
2. Sample a word $w$ according to $\phi_z$ (e.g. $w=$government)
PLSA – Model Inference using Expectation-Maximization (EM)

E-step: Fix $\phi, \theta$, estimate topic labels $z$ for every token in every document
M-step: Use estimated topic labels $z$ to estimate $\phi, \theta$
Guaranteed to converge to a stationary point, but not guaranteed optimal

Criticism of government response to the hurricane ...

Exact max likelihood is hard => approximate optimization with EM
How the EM Algorithm Works

Topic $\phi_1$
- government 0.3
- response 0.2

Topic $\phi_k$
- donate 0.1
- relief 0.05

Doc $\theta_1$
- .4
- .3
- .3

Doc $\theta_D$
- .2
- .5
- .3

Bayes rule:
\[
p(z = j \mid d, w) = \frac{p(z = j \mid d) p(w \mid z = j)}{\sum_{j' = 1}^{k} p(z = j' \mid d) p(w \mid z = j')} = \frac{\theta_{d,j} \phi_{j,w}}{\sum_{j' = 1}^{k} \theta_{d,j'} \phi_{j',w}}
\]
## Analysis of pLSA

<table>
<thead>
<tr>
<th>PROS</th>
<th>CONS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple, only one hyperparameter $k$</td>
<td>High model complexity $\rightarrow$ prone to overfitting</td>
</tr>
<tr>
<td>Easy to incorporate prior in the EM algorithm</td>
<td>The EM solution is neither optimal nor unique</td>
</tr>
</tbody>
</table>
Latent Dirichlet Allocation (LDA) [Blei et al. 02]

- Impose Dirichlet prior to the model parameters -> Bayesian version of pLSA

Generative process: First **generate** $\phi, \theta$ with Dirichlet prior, then generate each token in each document $d$ according to $\phi, \theta$

To mitigate overfitting
LDA – Model Inference

MAXIMUM LIKELIHOOD

- Aim to find parameters that maximize the likelihood
- Exact inference is intractable
- Approximate inference
  - Variational EM [Blei et al. 03]
  - Markov chain Monte Carlo (MCMC) – collapsed Gibbs sampler [Griffiths & Steyvers 04]

METHOD OF MOMENTS

- Aim to find parameters that fit the moments (expectation of patterns)
- Exact inference is tractable
  - Tensor orthogonal decomposition [Anandkumar et al. 12]
  - Scalable tensor orthogonal decomposition [Wang et al. 14a]
MCMC – Collapsed Gibbs Sampler
[Griffiths & Steyvers 04]

Sample each $z_i$ conditioned on $z_{-i}$

$$P(z_i = j \mid \mathbf{w}, \mathbf{z}_{-i}) \propto \frac{N_{w_j}^{(j)} + \beta}{N_j^{(j)} + V \beta} \cdot \frac{n_{d_i}^{(d_i)} + \alpha}{n_{d_i}^{(d_i)} + k \alpha}$$
Method of Moments
[Anandkumar et al. 12, Wang et al. 14a]

What parameters are most likely to generate the observed corpus?

What parameters fit the empirical moments?

Moments: expectation of patterns

<table>
<thead>
<tr>
<th></th>
<th>criticism</th>
<th>response</th>
<th>government</th>
</tr>
</thead>
<tbody>
<tr>
<td>length 1</td>
<td>0.03</td>
<td>0.01</td>
<td>0.04</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>criticism</th>
<th>response</th>
<th>government</th>
</tr>
</thead>
<tbody>
<tr>
<td>length 2 (pair)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>criticism response</td>
<td>0.001</td>
<td></td>
<td></td>
</tr>
<tr>
<td>criticism government</td>
<td>0.002</td>
<td></td>
<td></td>
</tr>
<tr>
<td>government response</td>
<td>0.003</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>criticism</th>
<th>response</th>
<th>government</th>
</tr>
</thead>
<tbody>
<tr>
<td>length 3 (triple)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>criticism government response</td>
<td>0.001</td>
<td></td>
<td></td>
</tr>
<tr>
<td>government response hurricane</td>
<td>0.005</td>
<td></td>
<td></td>
</tr>
<tr>
<td>criticism response hurricane</td>
<td>0.004</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Guaranteed Topic Recovery

**Theorem.** The patterns up to length 3 are sufficient for topic recovery

\[ M_2 = \sum_{j=1}^{k} \lambda_j \phi_j \otimes \phi_j, \quad M_3 = \sum_{j=1}^{k} \lambda_j \phi_j \otimes \phi_j \otimes \phi_j \]

\( \lambda_j \phi_j \) represents the weight and associated topic vector for each topic, where \( \lambda_j \) is the weight of topic \( j \) and \( \phi_j \) is the corresponding topic vector.

V: vocabulary size; \( k \): topic number

- **length 1**
  - criticism: 0.03
  - response: 0.01
  - government: 0.04

- **length 2 (pair)**
  - criticism response: 0.001
  - criticism government: 0.002
  - government response: 0.003

- **length 3 (triple)**
  - criticism government response: 0.001
  - government response hurricane: 0.005
  - criticism response hurricane: 0.004

- **length 3 (triple)**
Tensor Orthogonal Decomposition for LDA

Normalized pattern counts

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.03</td>
<td>AB: 0.001</td>
</tr>
<tr>
<td>B</td>
<td>0.01</td>
<td>BC: 0.002</td>
</tr>
<tr>
<td>C</td>
<td>0.04</td>
<td>AC: 0.003</td>
</tr>
</tbody>
</table>

V: vocabulary size
k: topic number

V_2

\[ V \]

\[ M_2 \]

\[ k \]

\[ \tilde{T} \]

\[ \phi_1 \]

government 0.3
response 0.2

\[ \phi_k \]

donate 0.1
relief 0.05

\[ \phi_1 \]

Input corpus

\[ V \]

\[ M_3 \]
Tensor Orthogonal Decomposition for LDA – Not Scalable

Input corpus

Normalized pattern counts

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.03</td>
<td>AB: 0.001</td>
</tr>
<tr>
<td>B</td>
<td>0.01</td>
<td>BC: 0.002</td>
</tr>
<tr>
<td>C</td>
<td>0.04</td>
<td>AC: 0.003</td>
</tr>
</tbody>
</table>

V: vocabulary size; k: topic number
L: # tokens; l: average doc length

Prohibitive to compute

Time: $O(V^3 k + Ll^2)$
Space: $O(V^3)$

table

government 0.3
response 0.2

donate 0.1
relief 0.05
Scalable Tensor Orthogonal Decomposition

Normalized pattern counts

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>AB</th>
<th>ABC</th>
<th>B</th>
<th>BC</th>
<th>ABD</th>
<th>C</th>
<th>AC</th>
<th>BCD</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.03</td>
<td>0.001</td>
<td>0.001</td>
<td>0.01</td>
<td>0.002</td>
<td>0.005</td>
<td>0.04</td>
<td>0.003</td>
<td>0.004</td>
</tr>
<tr>
<td>B</td>
<td>0.01</td>
<td></td>
<td></td>
<td>0.02</td>
<td></td>
<td></td>
<td>0.04</td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>0.04</td>
<td>0.003</td>
<td>0.004</td>
<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>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Sparse & low rank

\# nonzero $m \ll V^2$

Decomposable

Time: $O(Lk^2 + km)$

Space: $O(m)$

[WANG ET AL. 14A]
STOD is 20-3000 times faster

- And the recovery error is low when the sample is large enough
- Variance is almost 0

STOD – Scalable tensor orthogonal decomposition
TOD – Tensor orthogonal decomposition

<table>
<thead>
<tr>
<th>L=19M</th>
<th>L=39M</th>
</tr>
</thead>
<tbody>
<tr>
<td>dataset</td>
<td>loaded into memory</td>
</tr>
<tr>
<td></td>
<td>STOD</td>
</tr>
<tr>
<td>news</td>
<td>293</td>
</tr>
<tr>
<td>CS</td>
<td>541</td>
</tr>
</tbody>
</table>
Summary of LDA Model Inference

MAXIMUM LIKELIHOOD

- Approximate inference
  - slow, scan data thousands of times
  - large variance, no theoretic guarantee

- Numerous follow-up work
  - further approximation [Porteous et al. 08, Yao et al. 09, Hoffman et al. 12] etc.
  - parallelization [Newman et al. 09] etc.
  - online learning [Hoffman et al. 13] etc.

METHOD OF MOMENTS

- STOD [Wang et al. 14a]
  - fast, scan data twice
  - robust recovery with theoretic guarantee

New and promising!
Methodologies of Topic Mining

A. Traditional bag-of-words topic modeling

B. Extension of topic modeling
i) Flat -> hierarchical  
ii) Unigrams -> ngrams  
iii) Text -> text + entity

C. An integrated framework: CATHY
i) Recursive topic discovery  
ii) Phrase mining  
iii) Phrase and entity ranking
B. i) Flat Topics -> Hierarchical Topics

- In PLSA and LDA, a topic is selected from a flat pool of topics
- In hierarchical topic models, a topic is selected from a hierarchy

To generate a token in document $d$:
1. Sample a topic label $z$ according to $\theta_d$
2. Sample a word $w$ according to $\phi_z$
Hierarchical Topic Models

- Topics form a tree structure
  - nested Chinese Restaurant Process [Griffiths et al. 04]
  - recursive Chinese Restaurant Process [Kim et al. 12a]
  - LDA with Topic Tree [Wang et al. 14b]

- Topics form a DAG structure
  - Pachinko Allocation [Li & McCallum 06]
  - hierarchical Pachinko Allocation [Mimno et al. 07]
  - nested Chinese Restaurant Franchise [Ahmed et al. 13]
Hierarchical Topic Model Inference

MAXIMUM LIKELIHOOD

- Exact inference is intractable
- Approximate inference: variational inference or MCMC

Most popular

- Non recursive – all the topics are inferred at once

METHOD OF MOMENTS

- Scalable Tensor Recursive Orthogonal Decomposition [Wang et al. 14b]
  - fast and robust recovery with theoretic guarantee

- Recursive method - only for LDA with Topic Tree model
Recursive Inference for LDA with Topic Tree

- A large tree *subsumes* a smaller tree with shared model parameters

Flexible to decide when to terminate

Inference order

Easy to revise the tree structure

[WANG ET AL. 14B]
Scalable Tensor Recursive Orthogonal Decomposition

Input corpus + Topic t

Normalized pattern counts for t:

<table>
<thead>
<tr>
<th>A</th>
<th>AB</th>
<th>ABC</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.03</td>
<td>0.001</td>
<td>0.001</td>
</tr>
<tr>
<td>B</td>
<td>BC</td>
<td>ABD</td>
</tr>
<tr>
<td>0.01</td>
<td>0.002</td>
<td>0.005</td>
</tr>
<tr>
<td>C</td>
<td>AC</td>
<td>BCD</td>
</tr>
<tr>
<td>0.04</td>
<td>0.003</td>
<td>0.004</td>
</tr>
<tr>
<td>:</td>
<td>:</td>
<td>:</td>
</tr>
</tbody>
</table>

Topics:

- $\phi_{t/1}$
  - government: 0.3
  - response: 0.2
  - ...
- $\phi_{t/k}$
  - donate: 0.1
  - relief: 0.05
  - ...

[WANG ET AL. 14B]
B. ii) Unigrams -> N-Grams

- Motivation: unigrams can be difficult to interpret

The topic that represents the area of Machine Learning

<table>
<thead>
<tr>
<th>learning</th>
<th>support vector machines</th>
</tr>
</thead>
<tbody>
<tr>
<td>reinforcement</td>
<td>reinforcement learning</td>
</tr>
<tr>
<td>support</td>
<td>feature selection</td>
</tr>
<tr>
<td>machine</td>
<td>conditional random fields</td>
</tr>
<tr>
<td>vector</td>
<td>classification</td>
</tr>
<tr>
<td>selection</td>
<td>decision trees</td>
</tr>
<tr>
<td>feature</td>
<td></td>
</tr>
<tr>
<td>random</td>
<td></td>
</tr>
</tbody>
</table>
Various Strategies

- **Strategy 1**: generate bag-of-words -> generate sequence of tokens
  - Bigram topical model [Wallach 06], *topical n-gram model* [Wang et al. 07], phrase discovering topic model [Lindsey et al. 12]

- **Strategy 2**: post bag-of-words model inference, visualize topics with n-grams
  - Label topic [Mei et al. 07], TurboTopic [Blei & Lafferty 09], *KERT* [Danilevsky et al. 14]

- **Strategy 3**: prior bag-of-words model inference, mine phrases and impose to the bag-of-words model
  - Frequent pattern-enriched topic model [Kim et al. 12b], *TopMine* [El-kishky et al. 14]
Strategy 1 – Topical N-Gram Model & Phrase Discovering Topic Model

To generate a token in document $d$:

1. Sample a binary variable $x$ according to the previous token & topic label
2. Sample a topic label $z$ according to $\theta_d$
3. If $x = 0$ (new phrase), sample a word $w$ according to $\phi_z$; otherwise, sample a word $w$ according to $z$ and the previous token

High model complexity - overfitting  MCMC inference  High inference cost - slow

[WANG ET AL. 07, LINDSEY ET AL. 12]
Strategy 2 – Topical Keyphrase Extraction & Ranking (KERT)

- knowledge discovery using least squares support vector machine classifiers
- support vectors for reinforcement learning
- a hybrid approach to feature selection
- pseudo conditional random fields
- automatic web page classification in a dynamic and hierarchical way
- inverse time dependency in convex regularized learning
- postprocessing decision trees to extract actionable knowledge
- variance minimization least squares support vector machines
- ... 

Unigram topic assignment: Topic 1 & Topic 2

Topical keyphrase extraction & ranking

[DANILEVSKY ET AL. 14]
Framework of KERT

1. Run bag-of-words model inference, and assign topic label to each token

2. Extract candidate keyphrases within each topic

3. Rank the keyphrases in each topic
   ◦ Popularity: ‘information retrieval’ vs. ‘cross-language information retrieval’
   ◦ Discriminativeness: only frequent in documents about topic t
   ◦ Concordance: ‘active learning’ vs. ‘learning classification’
   ◦ Completeness: ‘vector machine’ vs. ‘support vector machine’

Comparability property: directly compare phrases of mixed lengths
Comparison of phrase ranking methods

The topic that represents the area of Machine Learning

<table>
<thead>
<tr>
<th>kpRel [Zhao et al. 11]</th>
<th>KERT (-popularity)</th>
<th>KERT (-discriminativeness)</th>
<th>KERT (-concordance)</th>
<th>KERT [Danilevsky et al. 14]</th>
</tr>
</thead>
<tbody>
<tr>
<td>learning</td>
<td>effective</td>
<td>support vector machines</td>
<td>learning</td>
<td>learning</td>
</tr>
<tr>
<td>classification</td>
<td>text</td>
<td>feature selection</td>
<td>classification</td>
<td>support vector machines</td>
</tr>
<tr>
<td>selection</td>
<td>probabilistic</td>
<td>reinforcement learning</td>
<td>selection</td>
<td>reinforcement learning</td>
</tr>
<tr>
<td>models</td>
<td>identification</td>
<td>conditional random fields</td>
<td>feature</td>
<td>feature selection</td>
</tr>
<tr>
<td>algorithm</td>
<td>mapping</td>
<td>constraint satisfaction</td>
<td>decision</td>
<td>conditional random fields</td>
</tr>
<tr>
<td>features</td>
<td>task</td>
<td>decision trees</td>
<td>bayesian</td>
<td>classification</td>
</tr>
<tr>
<td>decision</td>
<td>planning</td>
<td>dimensionality reduction</td>
<td>trees</td>
<td>decision trees</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Strategy 3 – Phrase Mining + Topic Model (ToPMine)

Strategy 2: the tokens in the same phrase may be assigned to different topics

knowledge discovery using least squares support vector machine classifiers…

→ Knowledge discovery and support vector machine should have coherent topic labels

Solution: switch the order of phrase mining and topic model inference

[knowledge discovery] using [least squares] [support vector machine] [classifiers] …

Phrase mining and document segmentation

[knowledge discovery] using [least squares] [support vector machine] [classifiers] …

Topic model inference with phrase constraints

More challenging than in strategy 2!

[EL-KISHKY ET AL. 14]
Phrase Mining: Frequent Pattern Mining + Statistical Analysis

\[
\alpha(A, B) = \frac{|AB| - |A||B|}{\sqrt{|AB|}}
\]

Significance score

Church et al. 91

Markov Blanket Feature Selection for Support Vector Machines.

Good Phrases
Phrase Mining: Frequent Pattern Mining + Statistical Analysis

Markov Blanket Feature Selection for Support Vector Machines.

**Significance score**

[Church et al. 91]

\[ \alpha(A, B) = \frac{|AB| - |A||B|/n}{\sqrt{|AB|}} \]

<table>
<thead>
<tr>
<th>[support vector machine]</th>
<th>90 ➔ 80</th>
</tr>
</thead>
<tbody>
<tr>
<td>[vector machine]</td>
<td>95 ➔ 0</td>
</tr>
<tr>
<td>[support vector]</td>
<td>100 ➔ 20</td>
</tr>
</tbody>
</table>

[Markov blanket] [feature selection] for [support vector machines]

[knowledge discovery] using [least squares]

[support vector machine] [classifiers]

…[support vector] for [machine learning]…
### PDLDA [Lindsey et al. 12] – Strategy 1 (3.72 hours)

<table>
<thead>
<tr>
<th>Topic 1</th>
<th>Topic 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>social networks</td>
<td>information retrieval</td>
</tr>
<tr>
<td>web search</td>
<td>text classification</td>
</tr>
<tr>
<td>time series</td>
<td>machine learning</td>
</tr>
<tr>
<td>search engine</td>
<td>support vector machines</td>
</tr>
<tr>
<td>management system</td>
<td>information extraction</td>
</tr>
<tr>
<td>real time</td>
<td>neural networks</td>
</tr>
<tr>
<td>decision trees</td>
<td>text categorization</td>
</tr>
</tbody>
</table>

### ToPMine [El-kishky et al. 14] – Strategy 3 (67 seconds)

<table>
<thead>
<tr>
<th>Topic 1</th>
<th>Topic 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>information retrieval</td>
<td>feature selection</td>
</tr>
<tr>
<td>social networks</td>
<td>machine learning</td>
</tr>
<tr>
<td>web search</td>
<td>semi supervised</td>
</tr>
<tr>
<td>search engine</td>
<td>large scale</td>
</tr>
<tr>
<td>information extraction</td>
<td>support vector machines</td>
</tr>
<tr>
<td>question answering</td>
<td>active learning</td>
</tr>
<tr>
<td>web pages</td>
<td>face recognition</td>
</tr>
</tbody>
</table>

---

Example Topical Phrases
<table>
<thead>
<tr>
<th>Topic 1</th>
<th>Topic 2</th>
<th>Topic 3</th>
<th>Topic 4</th>
<th>Topic 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>plant</td>
<td>church</td>
<td>palestinian</td>
<td>bush</td>
<td>drug</td>
</tr>
<tr>
<td>nuclear</td>
<td>catholic</td>
<td>israeli</td>
<td>house</td>
<td>aid</td>
</tr>
<tr>
<td>environmental</td>
<td>religious</td>
<td>israel</td>
<td>senate</td>
<td>health</td>
</tr>
<tr>
<td>energy</td>
<td>bishop</td>
<td>arab</td>
<td>year</td>
<td>hospital</td>
</tr>
<tr>
<td>year</td>
<td>pope</td>
<td>plo</td>
<td>bill</td>
<td>medical</td>
</tr>
<tr>
<td>waste</td>
<td>roman</td>
<td>army</td>
<td>president</td>
<td>patients</td>
</tr>
<tr>
<td>department</td>
<td>jewish</td>
<td>reported</td>
<td>congress</td>
<td>research</td>
</tr>
<tr>
<td>power</td>
<td>rev</td>
<td>west</td>
<td>tax</td>
<td>test</td>
</tr>
<tr>
<td>state</td>
<td>john</td>
<td>bank</td>
<td>budget</td>
<td>study</td>
</tr>
<tr>
<td>chemical</td>
<td>christian</td>
<td>state</td>
<td>committee</td>
<td>disease</td>
</tr>
<tr>
<td>n-grams</td>
<td>energy department</td>
<td>roman catholic</td>
<td>gaza strip</td>
<td>president bush</td>
</tr>
<tr>
<td></td>
<td>environmental protection agency</td>
<td>pope john paul</td>
<td>west bank</td>
<td>health care</td>
</tr>
<tr>
<td></td>
<td>nuclear weapons</td>
<td>john paul</td>
<td>palestine liberation organization</td>
<td>white house</td>
</tr>
<tr>
<td></td>
<td>acid rain</td>
<td>catholic church</td>
<td>united states</td>
<td>medical center</td>
</tr>
<tr>
<td></td>
<td>nuclear power plant</td>
<td>anti semitism</td>
<td>arab reports</td>
<td>united states</td>
</tr>
<tr>
<td></td>
<td>hazardous waste</td>
<td>baptist church</td>
<td>prime minister</td>
<td>house and senate</td>
</tr>
<tr>
<td></td>
<td>savannah river</td>
<td>united states</td>
<td>yitzhak shamir</td>
<td>members of congress</td>
</tr>
<tr>
<td></td>
<td>rocky flats</td>
<td>lutheran church</td>
<td>israel radio</td>
<td>defense secretary</td>
</tr>
<tr>
<td></td>
<td>nuclear power</td>
<td>episcopal church</td>
<td>occupied territories</td>
<td>capital gains tax</td>
</tr>
<tr>
<td></td>
<td>natural gas</td>
<td>church members</td>
<td>occupied west bank</td>
<td>pay raise</td>
</tr>
</tbody>
</table>

ToPMine: Experiments on Associate Press News (1989)
<table>
<thead>
<tr>
<th>Topic 1</th>
<th>Topic 2</th>
<th>Topic 3</th>
<th>Topic 4</th>
<th>Topic 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>unigrams</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>coffee</td>
<td>food</td>
<td>room</td>
<td>store</td>
<td>good</td>
</tr>
<tr>
<td>ice</td>
<td>good</td>
<td>parking</td>
<td>shop</td>
<td>food</td>
</tr>
<tr>
<td>cream</td>
<td>place</td>
<td>hotel</td>
<td>prices</td>
<td>place</td>
</tr>
<tr>
<td>flavor</td>
<td>ordered</td>
<td>stay</td>
<td>find</td>
<td>burger</td>
</tr>
<tr>
<td>egg</td>
<td>chicken</td>
<td>time</td>
<td>place</td>
<td>ordered</td>
</tr>
<tr>
<td>chocolate</td>
<td>roll</td>
<td>nice</td>
<td>buy</td>
<td>fries</td>
</tr>
<tr>
<td>breakfast</td>
<td>sushi</td>
<td>place</td>
<td>selection</td>
<td>chicken</td>
</tr>
<tr>
<td>tea</td>
<td>restaurant</td>
<td>great</td>
<td>items</td>
<td>tacos</td>
</tr>
<tr>
<td>cake</td>
<td>dish</td>
<td>area</td>
<td>love</td>
<td>cheese</td>
</tr>
<tr>
<td>sweet</td>
<td>rice</td>
<td>pool</td>
<td>great</td>
<td>time</td>
</tr>
<tr>
<td>n-grams</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ice cream</td>
<td>spring rolls</td>
<td>parking lot</td>
<td>grocery store</td>
<td>mexican food</td>
</tr>
<tr>
<td>iced tea</td>
<td>food was good</td>
<td>front desk</td>
<td>great selection</td>
<td>chips and salsa</td>
</tr>
<tr>
<td>french toast</td>
<td>fried rice</td>
<td>spring training</td>
<td>farmer’s market</td>
<td>food was good</td>
</tr>
<tr>
<td>hash browns</td>
<td>egg rolls</td>
<td>staying at the hotel</td>
<td>great prices</td>
<td>hot dog</td>
</tr>
<tr>
<td>frozen yogurt</td>
<td>chinese food</td>
<td>dog park</td>
<td>parking lot</td>
<td>rice and beans</td>
</tr>
<tr>
<td>eggs benedict</td>
<td>pad thai</td>
<td>room was clean</td>
<td>wal mart</td>
<td>sweet potato fries</td>
</tr>
<tr>
<td>peanut butter</td>
<td>dim sum</td>
<td>pool area</td>
<td>shopping center</td>
<td>pretty good</td>
</tr>
<tr>
<td>cup of coffee</td>
<td>thai food</td>
<td>great place</td>
<td>great place</td>
<td>carne asada</td>
</tr>
<tr>
<td>iced coffee</td>
<td>pretty good</td>
<td>staff is friendly</td>
<td>prices are reasonable</td>
<td>mac and cheese</td>
</tr>
<tr>
<td>scrambled eggs</td>
<td>lunch specials</td>
<td>free wifi</td>
<td>love this place</td>
<td>fish tacos</td>
</tr>
</tbody>
</table>
### Comparison of three strategies

<table>
<thead>
<tr>
<th>Strategy</th>
<th>ToPMine</th>
<th>PD-LDA</th>
<th>Turbo Topics</th>
<th>TNG</th>
<th>LDA</th>
<th>KERT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sampled dblp titles (k=5)</td>
<td>67(s)</td>
<td>3.72(hrs)</td>
<td>6.68(hrs)</td>
<td>146(s)</td>
<td>65(s)</td>
<td>68(s)</td>
</tr>
<tr>
<td>All dblp titles (k=30)</td>
<td>2.45(hrs)</td>
<td>~20.44(days)</td>
<td>&gt;30(days)*</td>
<td>5.57(hrs)</td>
<td>3.04(hrs)</td>
<td>3.08(hrs)</td>
</tr>
<tr>
<td>Sampled dblp abstracts (k=10)</td>
<td>340(s)</td>
<td>1.12(days)</td>
<td>&gt;10(days)*</td>
<td>853(s)</td>
<td>353(s)</td>
<td>1215(s)</td>
</tr>
<tr>
<td>All dblp abstracts (k=50)</td>
<td>10.88(hrs)</td>
<td>~95.9(days)</td>
<td>&gt;50(days)*</td>
<td>NA†</td>
<td>13.84(hours)</td>
<td>NA†</td>
</tr>
</tbody>
</table>

**Runtime**
- fastest
- slow
- slow
- fast

**Coherence of topics**
- strategy 3 > strategy 2 > strategy 1
Summary of Topical N-Gram Mining

- **Strategy 1:** generate bag-of-words -> generate sequence of tokens
  - integrated complex model; phrase quality and topic inference rely on each other
  - slow and overfitting

- **Strategy 2:** post bag-of-words model inference, visualize topics with n-grams
  - phrase quality relies on topic labels for unigrams
  - can be fast

- **Strategy 3:** prior bag-of-words model inference, mine phrases and impose to the bag-of-words model
  - topic inference relies on correct partition of documents, but not sensitive
  - can be fast
B. iii) Text Only -> Text + Entity

- What should be the output?
- How to use linked entity information?
Three Modeling Strategies

RESEMBLE ENTITIES TO DOCUMENTS

- An entity has a multinomial distribution over topics

<table>
<thead>
<tr>
<th>Surajit Chaudhuri</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.3</td>
<td>0.4</td>
</tr>
<tr>
<td>0.3</td>
<td>0.5</td>
</tr>
<tr>
<td>SIGMOD</td>
<td>0.2</td>
</tr>
</tbody>
</table>

RESEMBLE ENTITIES TO WORDS

- A topic has a multinomial distribution over each type of entities

  | KDD 0.3         | Jiawei Han 0.1 |
  | ICDM 0.2        | Christos Faloustos 0.05 |
  |                  | Over venues     |
  |                  | Over authors    |

RESEMBLE ENTITIES TO TOPICS

- An entity has a multinomial distribution over words

  | SIGMOD          |
  | database 0.3   |
  | system 0.2     |
  | ...            |
Resemble Entities to Documents

- Regularization - Linked documents or entities have similar topic distributions
  - iTopicModel [Sun et al. 09a]
  - TMBP-Regu [Deng et al. 11]

- Use entities as additional sources of topic choices for each token
  - Contextual focused topic model [Chen et al. 12] etc.

- Aggregate documents linked to a common entity as a pseudo document
  - Co-regularization of inferred topics under multiple views [Tang et al. 13]
Resemble Entities to Documents

- Regularization - Linked documents or entities have similar topic distributions

\[ \theta_2 \text{ should be similar to } \theta_1, \theta_3 \]

\[ \theta_1^d \text{ should be similar to } \theta_5^u, \theta_2^u, \theta_2^v \]

 iTopicModel [Sun et al. 09a]

TMBP-Regu [Deng et al. 11]
Resemble Entities to Documents

- Use entities as additional sources of topic choice for each token
  - Contextual focused topic model [Chen et al. 12]

To generate a token in document $d$:

1. Sample a variable $x$ for the context type
2. Sample a topic label $z$ according to $\theta$ of the context type decided by $x$
3. Sample a word $w$ according to $\phi_z$

On Random Sampling over Joins:

- $x = 1$, sample $z$ from document’s topic distribution
- $x = 2$, sample $z$ from author’s topic distribution
- $x = 3$, sample $z$ from venue’s topic distribution
Resemble Entities to Documents

- Aggregate documents linked to a common entity as a pseudo document
  - Co-regularization of inferred topics under multiple views [Tang et al. 13]
Three Modeling Strategies

RESEMBLE ENTITIES TO DOCUMENTS

- An entity has a multinomial distribution over topics

  
  | Surajit Chaudhuri | 0.3 | 0.4 | 0.3 |
  | SIGMOD            | 0.2 | 0.5 | 0.3 |

RESEMBLE ENTITIES TO WORDS

- A topic has a multinomial distribution over each type of entities

  
  | Topic 1 |
  | KDD 0.3  |
  | ICDM 0.2 |
  | Jiawei Han 0.1 |
  | Christos Faloustos 0.05 |
  | Over venues |
  | Over authors |

RESEMBLE ENTITIES TO TOPICS

- An entity has a multinomial distribution over words

  
  | SIGMOD |
  | database 0.3  |
  | system 0.2   |
  | ...         |
Resemble Entities to Topics

- Entity-Topic Model (ETM) [Kim et al. 12c]

To generate a token in document $d$:
1. Sample an entity $e$
2. Sample a topic label $z$ according to $\theta_d$
3. Sample a word $w$ according to $\phi_{z,e}$
Example topics learned by ETM

On a news dataset about Japan tsunami 2011

<table>
<thead>
<tr>
<th>$\phi_z$</th>
<th>$\phi_{z,e}$</th>
<th>$\phi_{z,e}$</th>
<th>$\phi_{z,e}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Relief Efforts</td>
<td>American Red Cross</td>
<td>Korea</td>
<td>Tokyo</td>
</tr>
<tr>
<td>japan</td>
<td>cross</td>
<td>japan</td>
<td>people</td>
</tr>
<tr>
<td>japanese</td>
<td>red</td>
<td>japanese</td>
<td>japan</td>
</tr>
<tr>
<td>people</td>
<td>japan</td>
<td>friends</td>
<td></td>
</tr>
<tr>
<td>tsunami</td>
<td>american</td>
<td>korea</td>
<td>japanese</td>
</tr>
<tr>
<td>earthquake</td>
<td>relief</td>
<td>korean</td>
<td>tokyo</td>
</tr>
<tr>
<td>disaster</td>
<td>support</td>
<td>donations</td>
<td>tsunami</td>
</tr>
<tr>
<td>world</td>
<td>donations</td>
<td>koreans</td>
<td>back</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$\phi_e$</th>
<th>$\phi_{z,e}$</th>
<th>$\phi_{z,e}$</th>
<th>$\phi_{z,e}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Naoto Kan</td>
<td>Relief Efforts</td>
<td>Nuclear Accident</td>
<td>Economic Effects</td>
</tr>
<tr>
<td>kan</td>
<td>bodies</td>
<td>kan</td>
<td>prime</td>
</tr>
<tr>
<td>minister</td>
<td>search</td>
<td>minister</td>
<td>rule</td>
</tr>
<tr>
<td>prime</td>
<td>kan</td>
<td>prime</td>
<td>bill</td>
</tr>
<tr>
<td>naoto</td>
<td>people</td>
<td>naoto</td>
<td>kan</td>
</tr>
<tr>
<td>government</td>
<td>troops</td>
<td>nuclear</td>
<td>powerful</td>
</tr>
<tr>
<td>tokyo</td>
<td>car</td>
<td>radiation</td>
<td>business</td>
</tr>
<tr>
<td>crisis</td>
<td>crisis</td>
<td>plant</td>
<td>minister</td>
</tr>
</tbody>
</table>
Three Modeling Strategies

RESEMBLE ENTITIES TO DOCUMENTS

- An entity has a multinomial distribution over topics

  - Surajit Chaudhuri: 0.3, 0.4, 0.3
  - ... (omitted)
  - SIGMOD: 0.2, 0.5, 0.3

RESEMBLE ENTITIES TO WORDS

- A topic has a multinomial distribution over each type of entities

  - Topic 1:
    - KDD: 0.3
    - ICDM: 0.2
    - ... (omitted)
  - Over venues
  - Jiawei Han: 0.1
    - ... (omitted)
  - Over authors
  - Christos Faloustos: 0.05
    - ... (omitted)

RESEMBLE ENTITIES TO TOPICS

- An entity has a multinomial distribution over words

  - SIGMOD:
    - database: 0.3
    - system: 0.2
    - ... (omitted)
Resemble Entities to Words

- Entities as additional elements to be generated for each doc
  - Conditionally independent LDA [Cohn & Hofmann 01]
  - CorrLDA1 [Blei & Jordan 03]
  - SwitchLDA & CorrLDA2 [Newman et al. 06]
  - NetClus [Sun et al. 09b]

To generate a token/entity in document $d$:
1. Sample a topic label $z$ according to $\theta_d$
2. Sample a token $w$ / entity $e$ according to $\phi_z$ or $\phi^e_z$
Comparison of Three Modeling Strategies for Text + Entity

RESEMBLE ENTITIES TO DOCUMENTS
- Entities regularize textual topic discovery

RESEMBLE ENTITIES TO WORDS
- Entities enrich and regularize the textual representation of topics

RESEMBLE ENTITIES TO TOPICS
- Each entity has its own profile

Examples:
- **Surajit Chaudhuri**
- **SIGMOD**

Topic 1:
- **KDD 0.3**
- **ICDM 0.2**
- **Jiawei Han 0.1**
- **Christos Faloustos 0.05**

Over venues:

Over authors:

# params = k*(E+V)

# params = k*E*V
Methodologies of Topic Mining

A. Traditional bag-of-words topic modeling

B. Extension of topic modeling
 i) Flat -> hierarchical
 ii) Unigrams -> ngrams
 iii) Text -> text + entity

C. An integrated framework: CATHY
 i) Recursive topic discovery
 ii) Phrase mining
 iii) Phrase and entity ranking
Break
C. An Integrated Framework

How to choose & integrate?

- **Hierarchy**
  - Sequence of tokens generative model
    - Strategy 1
  - Post inference, visualize topics with n-grams
    - Strategy 2
  - Prior inference, mine phrases and impose to the bag-of-words model
    - Strategy 3

- **Recursive**

- **Non recursive**
  - Resemble entities to documents
    - Modeling strategy 1
  - Resemble entities to topics
    - Modeling strategy 2
  - Resemble entities to words
    - Modeling strategy 3
C. An Integrated Framework

Compatible & effective

Hierarchy
Recursive
Non recursive

Sequence of tokens generative model
- Strategy 1
- Post inference, visualize topics with n-grams

Resemble entities to documents
- Modeling strategy 1

Resemble entities to topics
- Modeling strategy 2

Resemble entities to words
- Modeling strategy 3

Resemble entities to documents
- Strategy 1
- Strategy 2
- Strategy 3

Phrases
Entity
C. An Integrated Framework: Construct A Topical Hierarchy (CATHY)

- Hierarchy + phrase + entity

Input collection

i) Hierarchical topic discovery with entities

ii) Phrase mining

iii) Rank phrases & entities per topic

Output hierarchy with phrases & entities
Mining Framework – CATHY

Construct A Topical Hierarchy

Input collection

entity

text

i) Hierarchical topic discovery with entities

ii) Phrase mining

iii) Rank phrases & entities per topic

Output hierarchy with phrases & entities
Hierarchical Topic Discovery with Multi-Typed Entities [Wang et al. 13a,b]

- Resemble entities to words – every topic has a multinomial distribution over each type of entities

\[ \phi_1^1 \]
\[ \text{data} \ 0.2 \]
\[ \text{mining} \ 0.1 \]
\[ \ldots \]

\[ \phi_2^2 \]
Jiawei Han 0.1
Christos Faloustos 0.05
\[ \ldots \]

\[ \phi_3^3 \]
KDD 0.3
ICDM 0.2
\[ \ldots \]

\[ \phi_k^1 \]
\[ \text{database} \ 0.2 \]
\[ \text{system} \ 0.1 \]
\[ \ldots \]
\[ \text{words} \]

\[ \phi_k^2 \]
Surajit Chaudhuri 0.1
Jeff Naughton 0.05
\[ \ldots \]
\[ \text{authors} \]

\[ \phi_k^3 \]
SIGMOD 0.3
VLDB 0.3
\[ \ldots \]
\[ \text{venues} \]
Hierarchical Topic Discovery: Link Patterns

Computing machinery and intelligence

A.M. Turing

intelligence
computing
machinery

A.M. Turing
Hierarchical Topic Discovery: Link-Weighted Heterogeneous Network

A.M. Turing
intelligence
system
database
SIGMOD

word
author
venue
Hierarchical Topic Discovery: Generative Model for Link Patterns

- A single link has a latent topic path $z$

To generate a link between type $t_1$ and type $t_2$:
1. Sample a topic label $z$ according to $\theta$

Suppose $t_1 = t_2 = \text{word}$
Hierarchical Topic Discovery: Generative Model for Link Patterns

To generate a link between type $t_1$ and type $t_2$:
1. Sample a topic label $z$ according to $\theta$
2. Sample the first end node $u$ according to $\phi_{t_1}^z$

Suppose $t_1 = t_2 = \text{word}$
Hierarchical Topic Discovery: Generative Model for Link Patterns

To generate a link between type $t_1$ and type $t_2$:
1. Sample a topic label $z$ according to $\theta$.
2. Sample the first end node $u$ according to $\phi_{z}^{t_1}$.
3. Sample the second end node $v$ according to $\phi_{z}^{t_2}$.

Suppose $t_1 = t_2 = \text{word}$.
Hierarchical Topic Discovery: Generative Model for Link Patterns

Equivalently, we can generate # links between \( u \) and \( v \):

\[
e_{u,v} = e_{u,v}^1 + \cdots + e_{u,v}^k,\ e_{u,v}^z \sim \text{Poisson} \left( \theta_z \phi_{z,u}^{t_1} \phi_{z,v}^{t_2} \right)
\]

Suppose \( t_1 = t_2 = \text{word} \)
Recursive Model Inference Using Expectation-Maximization (EM)
Top-Down Recursion
Extension: Learn Link Type Importance

- Different link types may have different importance in topic discovery.

- Introduce a link type weight $\alpha_{x,y}$
  - Original link weight $e_{i,j}^{x,y,z} \rightarrow \alpha_{x,y}e_{i,j}^{x,y,z}$
  - $\alpha > 1$ – more important
  - $0 < \alpha < 1$ – less important

The EM solution is invariant to a constant scale up of all the link weights

we can assume w.l.o.g $\prod_{x,y} \alpha_{x,y}^{n_{x,y}} = 1$
Optimal Weight

\[ \alpha_{x,y} = \left[ \prod_{x,y} \left( \frac{1}{n_{x,y}} \sum_{i,j} e_{i,j}^{x,y,t} \log \frac{e_{i,j}^{x,y,t}}{s_{i,j}^{x,y,t}} \right) \right]^{n_{x,y}} \frac{1}{\sum_{x,y} n_{x,y}} \]

Average link weight

KL-divergence of prediction from observation
Learned importance of different link types

<table>
<thead>
<tr>
<th>Level</th>
<th>Word-word</th>
<th>Word-author</th>
<th>Author-author</th>
<th>Word-venue</th>
<th>Author-venue</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>.2451</td>
<td>.3360</td>
<td>.4707</td>
<td>5.7113</td>
<td>4.5160</td>
</tr>
<tr>
<td>2</td>
<td>.2548</td>
<td>.7175</td>
<td>.6226</td>
<td>2.9433</td>
<td>2.9852</td>
</tr>
</tbody>
</table>

Learned Link Importance & Topic Coherence
Phrase Mining

- Frequent pattern mining; no NLP parsing
- Statistical analysis for filtering bad phrases

i) Hierarchical topic discovery with entities
ii) Phrase mining
iii) Rank phrases & entities per topic

Output hierarchy with phrases & entities
## Examples of Mined Phrases

### News

<table>
<thead>
<tr>
<th>energy department</th>
<th>president bush</th>
</tr>
</thead>
<tbody>
<tr>
<td>environmental protection agency</td>
<td>white house</td>
</tr>
<tr>
<td>nuclear weapons</td>
<td>bush administration</td>
</tr>
<tr>
<td>acid rain</td>
<td>house and senate</td>
</tr>
<tr>
<td>nuclear power plant</td>
<td>members of congress</td>
</tr>
<tr>
<td>hazardous waste</td>
<td>defense secretary</td>
</tr>
<tr>
<td>savannah river</td>
<td>capital gains tax</td>
</tr>
</tbody>
</table>

### Computer science

<table>
<thead>
<tr>
<th>information retrieval</th>
<th>feature selection</th>
</tr>
</thead>
<tbody>
<tr>
<td>social networks</td>
<td>machine learning</td>
</tr>
<tr>
<td>web search</td>
<td>semi supervised</td>
</tr>
<tr>
<td>search engine</td>
<td>large scale</td>
</tr>
<tr>
<td>information extraction</td>
<td>support vector machines</td>
</tr>
<tr>
<td>question answering</td>
<td>active learning</td>
</tr>
<tr>
<td>web pages</td>
<td>face recognition</td>
</tr>
</tbody>
</table>


Phrase & Entity Ranking

- Ranking criteria: popular, discriminative, concordant

1. Hierarchical topic discovery w/ entities
2. Phrase mining
3. Rank phrases & entities per topic

Output hierarchy w/ phrases & entities
Phrase & Entity Ranking – Estimate Topical Frequency

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Total</th>
<th>ML</th>
<th>DB</th>
<th>DM</th>
<th>iR</th>
</tr>
</thead>
<tbody>
<tr>
<td>support vector machines</td>
<td>85</td>
<td>85</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>query processing</td>
<td>252</td>
<td>0</td>
<td>212</td>
<td>27</td>
<td>12</td>
</tr>
<tr>
<td>Hui Xiong</td>
<td>72</td>
<td>0</td>
<td>0</td>
<td>66</td>
<td>6</td>
</tr>
<tr>
<td>SIGIR</td>
<td>2242</td>
<td>444</td>
<td>378</td>
<td>303</td>
<td>1117</td>
</tr>
</tbody>
</table>

E.g.

\[
p(z = DB \mid \text{query processing}) = \frac{p(z=DB)p(\text{query} \mid z=DB)p(\text{processing} \mid z=DB)}{\sum_t p(z=t)p(\text{query} \mid z=t)p(\text{processing} \mid z=t)} = \frac{\theta_{DB \Phi_{DB,query}} \Phi_{DB,processing}}{\sum_t \theta_t \Phi_{t,query} \Phi_{t,processing}}
\]

Frequent pattern mining

Estimated by Bayes rule
Phrase & Entity Ranking – Ranking Function

- ‘Popular’ indicator of phrase or entity $A$ in topic $t$: $p(A|t)$
- ‘Discriminative’ indicator of phrase or entity $A$ in topic $t$: $\log \frac{p(A|t)}{p(A|T)}$
- ‘Concordance’ indicator of phrase $A$: $\alpha(A) = \frac{|A| - E(|A|)}{\text{std}(|A|)}$

$$r_t(A) = p(A|t) \log \frac{p(A|t)}{p(A|T)} + \omega \alpha(A)$$

$T$: topic for comparison

Significance score used for phrase mining

Pointwise KL-divergence
Evaluation: Intrusion detection [Chang et al. 09]

Example topics: database & information retrieval
**Evaluation method - Intrusion detection**

### Question 1/80
**Parent topic**
- database systems
- data management
- query processing
- management system
- data system

**Child topic 1**
- web search
- search engine
- semantic web
- search results
- web pages

**Child topic 2**
- data management
- data integration
- data sources
- data warehousing
- data applications

**Child topic 3**
- query processing
- query optimization
- query databases
- relational databases
- query data

**Child topic 4**
- database system
- database design
- expert system
- management system
- design system

---

### Which child topic does not belong to the given parent topic?

### Question 1/130
**Phrase Intrusion**
- data mining
- association rules
- logic programs
- data streams

---

### Which phrase does not belong?

<table>
<thead>
<tr>
<th>Phrase 1</th>
<th>Phrase 2</th>
<th>Phrase 3</th>
<th>Phrase 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>normal</td>
<td>normal</td>
<td>normal</td>
<td>normal</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>intruder</td>
<td>intruder</td>
<td>intruder</td>
<td>intruder</td>
</tr>
</tbody>
</table>

---

### Question 2/130
**Phrase Intrusion**
- natural language
- query optimization
- data management
- database systems
Ranked phrases + entities > unigrams
Applications: entity, community profiling...
Important research areas in SIGIR conference?

**SIGIR (2,432 papers)**

- **ML (443.8)**
  - support vector machines
  - collaborative filtering
  - text categorization
  - text classification
  - conditional random fields

- **DB (377.7)**
  - information systems
  - artificial intelligence
  - distributed information retrieval
  - query evaluation

- **DM (302.7)**
  - event detection
  - large collections
  - similarity search
  - duplicate detection
  - large scale

- **IR (1,117.4)**
  - information retrieval
  - question answering
  - web search
  - natural language processing
  - document retrieval

**Other research areas**

- **ML (108.9)**
  - matrix factorization
  - hidden Markov models
  - maximum entropy
  - link analysis
  - non-negative matrix factorization

- **DB (160.3)**
  - text categorization
  - text classification
  - document clustering
  - multi-document summarization
  - naïve Bayes

- **DM (583.0)**
  - information retrieval
  - question answering
  - relevance feedback
  - document retrieval
  - ad hoc

- **IR (260.0)**
  - web search
  - search engine
  - search results
  - world wide web
  - web search results

- **Others (127.3)**
  - word sense disambiguation
  - named entity recognition
  - domain knowledge
  - dependency parsing
Outline

1. Introduction to mining latent entity structures
2. Mining latent topic hierarchies
3. Mining latent entity relations
4. Mining latent entity concepts
5. Trends and research problems
Discovery of Hidden Entity Relations

Topics

Unstructured text & linked entities -> structured hierarchies

Relations

Discover latent relations between entities

entity

text
Example Hidden Relations

- **Academic family** from research publications
  - Jeff Ullman
  - Surajit Chaudhuri (1991)
  - Jeffrey Naughton (1987)

- **Social relationship** from online social network

  - Alumni
  - Colleague
  - Club friend

Jeffrey Naughton (1987)
Entity Relation Mining

A. Representative mining paradigms

B. Patterns, rules and constraints for reasoning

C. Methodologies for dependency modeling
Mining Paradigms

A. Representative mining paradigms
   ◦ Similarity search of relationships
   ◦ Classify or cluster entity relationships
   ◦ Slot filling

B. Patterns, rules and constraints for reasoning

C. Methodologies for dependency modeling
Similarity Search of Relationships

- Input: relation instance
- Output: relation instances with similar semantics

(Jeff Ullman, Surajit Chaudhuri) ➔ (Jeffrey Naughton, Joseph M. Hellerstein)
  Is advisor of

(Apple, iPad) ➔ (Microsoft, Surface)
  Produce tablet

(Jiawei Han, Chi Wang)

...
Classify or Cluster Entity Relationships

- Input: relation instances with unknown relationship
- Output: predicted relationship or clustered relationship

(Jeff Ullman, Surajit Chaudhuri)

(Jeff Ullman, Hector Garcia)

- Is advisor of
- Is colleague of

Alumni
Colleague
Club friend
Slot Filling

- Input: relation instance with a missing element (slot)
- Output: fill the slot

is advisor of (?, Surajit Chaudhuri) → Jeff Ullman
produce tablet (Apple, ?) → iPad

<table>
<thead>
<tr>
<th>Model</th>
<th>Brand</th>
</tr>
</thead>
<tbody>
<tr>
<td>S80</td>
<td>?</td>
</tr>
<tr>
<td>A10</td>
<td>?</td>
</tr>
<tr>
<td>T1460</td>
<td>?</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Model</th>
<th>Brand</th>
</tr>
</thead>
<tbody>
<tr>
<td>S80</td>
<td>Nikon</td>
</tr>
<tr>
<td>A10</td>
<td>Canon</td>
</tr>
<tr>
<td>T1460</td>
<td>Benq</td>
</tr>
</tbody>
</table>
Patterns, Rules & Constraints for Reasoning

A. Representative mining paradigms
   ◦ Similarity search of relationships
   ◦ Classify or cluster entity relationships
   ◦ Slot filling

B. Patterns, rules and constraints for reasoning
   ◦ Text patterns
   ◦ Linkage patterns
   ◦ Dependency rules and constraints

C. Methodologies for dependency modeling
Text Patterns

- Syntactic patterns
  - [Bunescu & Mooney 05b]
  - The headquarters of Google are situated in Mountain View

- Dependency parse tree patterns
  - [Zelenko et al. 03]
  - [Culotta & Sorensen 04]
  - [Bunescu & Mooney 05a]
  - Jane says John heads XYZ Inc.

- Topical patterns
  - [McCallum et al. 05] etc.

<table>
<thead>
<tr>
<th>Topic 5 “Grant Proposals”</th>
<th>Topic 31 “Meeting Setup”</th>
<th>Topic 38 “ML Models”</th>
<th>Topic 41 “Friendly Discourse”</th>
</tr>
</thead>
<tbody>
<tr>
<td>proposal</td>
<td>today</td>
<td>model</td>
<td>great</td>
</tr>
<tr>
<td>data</td>
<td>tomorrow</td>
<td>models</td>
<td>good</td>
</tr>
<tr>
<td>budget</td>
<td>time</td>
<td>inference</td>
<td>don</td>
</tr>
<tr>
<td>work</td>
<td>ll</td>
<td>conditional</td>
<td>sounds</td>
</tr>
<tr>
<td>year</td>
<td>meeting</td>
<td>methods</td>
<td>work</td>
</tr>
<tr>
<td>glenn</td>
<td>week</td>
<td>number</td>
<td>wishes</td>
</tr>
<tr>
<td>nsf</td>
<td>talk</td>
<td>sequence</td>
<td>talk</td>
</tr>
<tr>
<td>project</td>
<td>meet</td>
<td>learning</td>
<td>interesting</td>
</tr>
<tr>
<td>sets</td>
<td>morning</td>
<td>graphical</td>
<td>time</td>
</tr>
<tr>
<td>support</td>
<td>monday</td>
<td>random</td>
<td>hear</td>
</tr>
</tbody>
</table>
Linkage Patterns - Topology

- Edge sign prediction [Leskovec et al. 10]
  - Epinions: Trust/Distrust
  - Wikipedia: Support/Oppose
  - Slashdot: Friend/Foe

Triad counts: counts of signed triads edge u->v takes part in
Linkage Patterns - Traffic

- Manager-subordinate relationship from email exchange [Diehl et al. 07]

Possible communication events for relationship \((n_a, n_b)\)

<table>
<thead>
<tr>
<th>From</th>
<th>Recipients Include</th>
<th>From</th>
<th>Recipients Include</th>
</tr>
</thead>
<tbody>
<tr>
<td>(n_a)</td>
<td>(n_b)</td>
<td>(n_b)</td>
<td>(n_a)</td>
</tr>
<tr>
<td>(n_a)</td>
<td>(n_c) and not (n_b)</td>
<td>(n_b)</td>
<td>(n_c) and not (n_a)</td>
</tr>
<tr>
<td>(n_c)</td>
<td>(n_a) and not (n_b)</td>
<td>(n_c)</td>
<td>(n_b) and not (n_a)</td>
</tr>
<tr>
<td>(n_c)</td>
<td>(n_a) and (n_b)</td>
<td>(n_c)</td>
<td>(n_a) and (n_b)</td>
</tr>
</tbody>
</table>

- # msgs
- quartiles for # recipients
Linkage Patterns – Traffic Dynamics

- Advisor-advisee relationship from research publications [Wang et al. 10]

A (papers by the candidate advisor)

\[ \text{kulc}(A, B) = \frac{|A \cap B|}{2} \left( \frac{1}{|A|} + \frac{1}{|B|} \right) \]

Kulczinski

\[ IR(A, B) = \frac{|A| - |B|}{|A \cup B|} \]

Imbalance ratio

Coauthorship trend over time

Kulczinski

Imbalance Ratio
Dependency Rules & Constraints (Advisor-Advisee Relationship)

E.g., role transition - one cannot be advisor before graduation
Dependency Rules & Constraints (Social Relationship)

**ATTRIBUTE-RELATIONSHIP**

*Friends of the same relationship type share the same value for only certain attribute*

**CONNECTION-RELATIONSHIP**

*The friends having different relationships are loosely connected*
Methodologies for Dependency Modeling

- **Factor graph**
  - [Wang et al. 10, 11, 12]
  - [Tang et al. 11]

- **Optimization framework**
  - [McAuley & Leskovec 12]
  - [Li, Wang & Chang 14]

- **Graph-based ranking**
  - [Yakout et al. 12]
Factor Graph [Wang et al. 10, 11, 12]

- Factor graph – model the joint likelihood by multiple factors
- Each factor is a potential function defined on a few variables
  - $y_x$ - $a_x$’s advisor. E.g., $y_2 \in \{0,1\}, y_4 \in \{0,3\}$
  - $g_x(y_x)$ - traffic dynamic; $f_x(y_x, C_x)$ - role transition

$$P = \frac{1}{Z} \prod_{\text{author } a_x} g_x(y_x) f_x(y_x, C_x)$$

1 if all the constraints are satisfied; 0 otherwise

average of Kulczynski measure and IR measure
Factor Graph - Inference

- Traditional inference algorithm message passing [Koller & Friedman 09]
  - Exact inference: sum-product + junction tree
  - Approximate inference: loopy belief propagation

- More optimized message passing algorithm [Wang et al. 10]
  - Specialized schedule according to the DAG structure
  - Eliminate function nodes and internal messages
  - Reduce the message from a vector to a scalar

Not scalable. Fails to finish for thousands of variables

Flooding schedule; causing repetitive information flow

Standard message passing algorithm
message is a marginal function (a vector)

Our message passing algorithm
message is a scalar
After the message passing, we collect the message to find the most likely advisor.

\[ r_{ij} = \max P(y_i = j) = \exp (\text{sent}_{ij} + \text{recv}_{ij}) \]

how likely \( a_j \) is \( a_i \)'s advisor
Experiment Results

- DBLP data: 654,628 authors, 1,076,946 publications, publishing time provided
- Labeled data: Math Genealogy; AI Genealogy; Faculty Homepage

<table>
<thead>
<tr>
<th>Datasets</th>
<th>RULE</th>
<th>SVM</th>
<th>Our method</th>
</tr>
</thead>
<tbody>
<tr>
<td>TEST1</td>
<td>69.9%</td>
<td>73.4%</td>
<td>84.4%</td>
</tr>
<tr>
<td>TEST2</td>
<td>69.8%</td>
<td>74.6%</td>
<td>84.3%</td>
</tr>
<tr>
<td>TEST3</td>
<td>80.6%</td>
<td>86.7%</td>
<td>91.3%</td>
</tr>
</tbody>
</table>

Accuracy of relationship prediction

- Heuristic rule
- Support vector machine (no role transition)
The proposed algorithm is efficient and effective.

<table>
<thead>
<tr>
<th>Advisee</th>
<th>Top Ranked Advisor</th>
<th>Time</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>David M. Blei</td>
<td>1. Michael I. Jordan</td>
<td>01-03</td>
<td>PhD advisor, 2004 grad</td>
</tr>
<tr>
<td></td>
<td>2. John D. Lafferty</td>
<td>05-06</td>
<td>Postdoc, 2006</td>
</tr>
<tr>
<td>Hong Cheng</td>
<td>1. Qiang Yang</td>
<td>02-03</td>
<td>MS advisor, 2003</td>
</tr>
<tr>
<td></td>
<td>2. Jiawei Han</td>
<td>04-08</td>
<td>PhD advisor, 2008</td>
</tr>
<tr>
<td>Sergey Brin</td>
<td>1. Rajeev Motawani</td>
<td>97-98</td>
<td>“Unofficial advisor”</td>
</tr>
</tbody>
</table>
Other examples of hierarchical relationship

Reply relationship in online discussion  [Wang et al. 11]
Other examples of hierarchical relationship

Family tree

[Wang et al. 12]
Solution in the generic setting

A large number of factors may exist

Their importance is unknown

Step 1: create a candidate DAG according to certain expectation about a partial order among the objects

Step 2: model the dependency of relations with a factor graph

Step 3: learn the importance of different expectations by Maximum A Posterior (MAP) inference

Step 4: maximize the joint likelihood and predict true relationship
### Categorization of potential functions

*See other types of potentials in [Wang et al. 12]*

<table>
<thead>
<tr>
<th>Type</th>
<th>Cognitive description</th>
<th>Potential definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Homophile</strong></td>
<td>Parent and child are similar</td>
<td>$g(y_i) = \text{sim}(v_i, v_{y_i})$</td>
</tr>
<tr>
<td><strong>Polarity</strong></td>
<td>Parent is superior to child</td>
<td>$g(y_i) = \text{asim}(v_i \rightarrow v_{y_i})$</td>
</tr>
</tbody>
</table>

#### Singleton potentials

- **Type:** Categorization of potential functions
- **Potential definition:**
  - $g(y_i) = \text{sim}(v_i, v_{y_i})$

#### Pairwise potentials

- **Type:** Categorization of potential functions
- **Potential definition:**
  - $f(y_i, y_j) = -|CP(v_i, v_j, v_{y_i}, v_{y_j})|$

- E.g., role transition
Optimization Framework [Li et al. 14]

- Relationship in online social networks

Some users provide attributes in their online profiles
Some users’ attributes are missing
Optimization Framework to Model the Dual Dependency

- Design cost function to capture the dependences
- Find the unknown variables that minimize the cost function

Observed User Connections

Partially Observed User Attributes

Unobserved Friends’ circles

The friends having different relationships are loosely connected

Friends of the same relationship type share the same value for only certain attribute
Co-Proﬁling of Relationship & Attributes

\[
\lambda_1 \sum_{t=1}^{K} \sum_{e_{ij} \in E, v_i, v_j \in C_i} [w_i(f_i - f_j)]^2 + \lambda_2 \sum_{t=1}^{K} \sum_{v_i \in L \cap C_i} (w_i f_i - 1)^2 + \lambda_3 \sum_{e_{ij} \in E', x_i \neq x_j} 1
\]

**Attribute-Relationship Dependence**

\[w_2 = <0, 0, 0, 1, 1, 0>\]

**Circle 1:** friends likely to share employee

**Circle 2:** friends likely to share college

\[f_4 = <0, 1, 0, 0, 1, 0, 0.1>\]

**Connection-Relationship Dependence**

**Association Vector** \[w_1 = <1, 1, 0, 0, 0, 0>\]

**Attribute Vector** \[f_1 = <1, 0, 0, 1, 0, 0, 0.1>\]

**Circle Assignment** \[x_i = 1\]

Yahoo | Twitter | Google | Berkeley | Stanford | UIUC | ...
--- | --- | --- | --- | --- | --- | ---

Yahoo | Twitter | Google | Berkeley | Stanford | UIUC | ...
Inference Algorithm

\[ \lambda_1 \sum_{t=1}^{K} \sum_{e_{ij} \in E, v_i, v_j \in C_i} [w_i(f_i - f_j)]^2 + \lambda_2 \sum_{t=1}^{K} \sum_{v_i \in L \cap C_i} (w_i f_i - 1)^2 + \lambda_3 \sum_{e_{ij} \in E', x_i \neq x_j} 1 \]

1. Update User Attribute Vectors $f_i$
   - Only propagate values from friends in the same circles
   - Only propagate the attribute value associated with the circle

2. Update User Circle Assignments $x_i, C_i$
   - Consider both user’s attributes and connections

3. Update Circle Association Vectors $w_i$
   - Make association vector sparse
Coprofiling of relationship + attributes outperforms baselines for both

LinkedIn data

<table>
<thead>
<tr>
<th>Users</th>
<th>Connections</th>
</tr>
</thead>
<tbody>
<tr>
<td>19K</td>
<td>110K</td>
</tr>
</tbody>
</table>

8K connections are labeled
Methodologies for Dependency Modeling

- **Factor graph**
  - [Wang et al. 10, 11, 12]
  - [Tang et al. 11]

- **Optimization framework**
  - [McAuley & Leskovec 12]
  - [Li, Wang & Chang 14]

- **Graph-based ranking**
  - [Yakout et al. 12]

- Suitable for discrete variables
- Probabilistic model with general inference algorithms
- Both discrete and real variables
- Special optimization algorithm needed
- Similar to PageRank
- Suitable when the problem can be modeled as ranking on graphs
Outline

1. Introduction to mining latent entity structures
2. Mining latent topic hierarchies
3. Mining latent entity relations
4. Mining latent entity concepts
5. Trends and research problems
Mining Entity Concepts for Typing

- **Topics**
  - Unstructured text & linked entities -> structured hierarchies

- **Concepts**
  - Discover the type of entities and disambiguate their mentions

- **Relations**
  - Discover latent relations between entities
# Concepts as Entity Types

- Top 10 active politicians regarding healthcare issues?
- Influential high-tech companies in Silicon Valley?

<table>
<thead>
<tr>
<th>Concept</th>
<th>Entity</th>
<th>Mention</th>
</tr>
</thead>
<tbody>
<tr>
<td>politician</td>
<td>Barack Obama</td>
<td><em>Obama says more than 6M signed up for health care...</em></td>
</tr>
<tr>
<td>high-tech company</td>
<td>Apple</td>
<td><em>Apple leads in list of Silicon Valley's most-valuable brands...</em></td>
</tr>
</tbody>
</table>
Source of Concepts, Entities & Mentions

<table>
<thead>
<tr>
<th>Concept</th>
<th>Entity</th>
<th>Mention</th>
</tr>
</thead>
<tbody>
<tr>
<td>politician</td>
<td>Barack Obama</td>
<td><em>Obama</em> says more than 6M signed up for health care...</td>
</tr>
<tr>
<td>high-tech company</td>
<td>Apple</td>
<td><em>Apple</em> leads in list of Silicon Valley's most-valuable brands...</td>
</tr>
</tbody>
</table>
Mining Entity Concepts for Typing

- Large scale taxonomies
  - Built from human-curated knowledgebases
  - Constructed from free text

- Type entities in text

- Type entities in web tables and lists
## Large Scale Taxonomies

<table>
<thead>
<tr>
<th>Name</th>
<th>Source</th>
<th># concepts</th>
<th># entities</th>
<th>Hierarchy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dbpedia (v3.9)</td>
<td>Wikipedia infoboxes</td>
<td>529</td>
<td>3M</td>
<td>Tree</td>
</tr>
<tr>
<td>YAGO2s</td>
<td>Wiki, WordNet, GeoNames</td>
<td>350K</td>
<td>10M</td>
<td>Tree</td>
</tr>
<tr>
<td>Freebase</td>
<td>Miscellaneous</td>
<td>23K</td>
<td>23M</td>
<td>Flat</td>
</tr>
<tr>
<td>Probase (MS.KB)</td>
<td>Web text</td>
<td>2M</td>
<td>5M</td>
<td>DAG</td>
</tr>
</tbody>
</table>

### Arnold Schwarzenegger

- **Types**: Person (People), US Politician (Government), Film actor (Film), Film producer (Film), Pro Athlete (Sports), Sports Award Winner (Sports)
- **Also known as**: Arnold Alois Schwarzenegger, The Governor
- **Gender**: Male
- **Date of Birth**: Jul 30, 1947
- **Place of Birth**: Thal, Austria
- **Country Of Nationality**: United States
- **Profession**: Politician, Bodybuilder, Entrepreneur, Actor

![Diagram of YAGO2s](image)
Relying on knowledgebases – entity linking
- Context similarity: [Bunescu & Pascal 06] etc.
- Topical coherence: [Cucerzan 07] etc.
- Context similarity + entity popularity + topical coherence: Wikifier [Ratinov et al. 11]
- Jointly linking multiple mentions: AIDA [Hoffart et al. 11] etc.
- ...

The AAAI organization recently announced that Michael Jordan is newly elected as AAAI fellow.
Limitation of Entity Linking

- Low recall of knowledgebases
- Sparse concept descriptors

Can we type entities without relying on knowledgebases?

Yes! Exploit the redundancy in the corpus

- Not relying on knowledgebases: targeted disambiguation of ad-hoc, homogeneous entities [Wang et al. 12]
- Partially relying on knowledgebases: mining additional evidence in the corpus for disambiguation [Li et al. 13]
Targeted Disambiguation
[Wang et al. 12]

<table>
<thead>
<tr>
<th>Entity Id</th>
<th>Entity Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>e1</td>
<td>Microsoft</td>
</tr>
<tr>
<td>e2</td>
<td>Apple</td>
</tr>
<tr>
<td>e3</td>
<td>HP</td>
</tr>
</tbody>
</table>

**Microsoft** and **Apple** are the developers of three of the most popular operating systems.

**Microsoft**’s new operating system, Windows 8, is a PC operating system for the tablet age...

**Apple** trees take four to five years to produce their first fruit...

CEO Meg Whitman said that **HP** is focusing on Windows 8 for its tablet strategy.

Audi is offering a racing version of its hottest TT model: a 380 **HP**, front-wheel...
Targeted Disambiguation

Target entities

<table>
<thead>
<tr>
<th>Entity Id</th>
<th>Entity Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>e1</td>
<td>Microsoft</td>
</tr>
<tr>
<td>e2</td>
<td>Apple</td>
</tr>
<tr>
<td>e3</td>
<td>HP</td>
</tr>
</tbody>
</table>

- **d1**: Microsoft’s new operating system, Windows 8, is a PC operating system for the tablet age ...
- **d2**: Microsoft and Apple are the developers of three of the most popular operating systems
- **d3**: Apple trees take four to five years to produce their first fruit...
- **d4**: CEO Meg Whitman said that HP is focusing on Windows 8 for its tablet strategy
- **d5**: Audi is offering a racing version of its hottest TT model: a 380 HP, front-wheel ...
Microsoft and Apple are the developers of three of the most popular operating systems.

Apple trees take four to five years to produce their first fruit...

CEO Meg Whitman said that HP is focusing on Windows 8 for its tablet strategy.

Audi is offering a racing version of its hottest TT model: a 380 HP, front-wheel...
Microsoft and Apple are the developers of three of the most popular operating systems. Apple trees take four to five years to produce their first fruit. CEO Meg Whitman said that HP is focusing on Windows 8 for its tablet strategy. Audi is offering a racing version of its hottest TT model: a 380 HP, front-wheel...
Microsoft’s new operating system, Windows 8, is a PC operating system for the tablet age...

Microsoft and Apple are the developers of three of the most popular operating systems

Apple trees take four to five years to produce their first fruit...

CEO Meg Whitman said that HP is focusing on Windows 8 for its tablet strategy

Audi is offering a racing version of its hottest TT model: a 380 HP, front-wheel...
Insight – Leverage Homogeneity

- **Hypothesis**: the context between two true mentions is more similar than between two false mentions across two distinct entities, as well as between a true mention and a false mention.

- **Caveat**: the context of false mentions can be similar among themselves within an entity.
Microsoft and Apple are the developers of three of the most popular operating systems.

Microsoft’s new operating system, Windows 8, is a PC operating system for the tablet age...

Microsoft and Apple are the developers of three of the most popular operating systems.

Apple trees take four to five years to produce their first fruit...

CEO Meg Whitman said that HP is focusing on Windows 8 for its tablet strategy.

Audi is offering a racing version of its hottest TT model: a 380 HP, front-wheel...
Microsoft’s new operating system, Windows 8, is a PC operating system for the tablet age…

Microsoft and Apple are the developers of three of the most popular operating systems.

Apple trees take four to five years to produce their first fruit…

CEO Meg Whitman said that HP is focusing on Windows 8 for its tablet strategy.

Audi is offering a racing version of its hottest TT model: a 380 HP, front-wheel…
Microsoft’s new operating system, Windows 8, is a PC operating system for the tablet age...

Microsoft and Apple are the developers of three of the most popular operating systems.

Apple trees take four to five years to produce their first fruit...

CEO Meg Whitman said that HP is focusing on Windows 8 for its tablet strategy.

Audi is offering a racing version of its hottest TT model: a 380 HP, front-wheel...
Microsoft’s new operating system, Windows 8, is a PC operating system for the tablet age...

Microsoft and Apple are the developers of three of the most popular operating systems.

Apple trees take four to five years to produce their first fruit...

CEO Meg Whitman said that HP is focusing on Windows 8 for its tablet strategy.

Audi is offering a racing version of its hottest TT model: a 380 HP, front-wheel...
### MentionRank

<table>
<thead>
<tr>
<th>Entity Id</th>
<th>Entity Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>e1</td>
<td>Microsoft</td>
</tr>
<tr>
<td>e2</td>
<td>Apple</td>
</tr>
</tbody>
</table>

- $e_1$ and $d_1$ linked by $r_{11}$.
- $e_2$ and $d_2$ linked by $r_{21}$ and $r_{22}$.
- $e_2$ and $d_3$ unlinked.

Diagram showing the relationships between entities and documents.
MentionRank

- $r_{ij} \in [0,1]$ – confidence for $(e_i, d_j)$ to be a true mention.

- $\pi_{ij} \in [0,1]$ – the prior estimate of $(e_i, d_j)$ to be a true mention (co-mention)

- $\mu_{ij, i'j'} \in [0,1]$ – the weight between $(e_i, d_j)$ and $(e_{i'}, d_{j'})$ (context similarity)
Experiment results

A Pleasure to Burn, Golden Reflections, Pathfinder...
Java, Python, Ruby, Perl, R...

Science Fiction Books
Programming Languages
Type Entities in Web Tables & Lists

[Limaye et al. 10]
## Type Entities in Web Tables & Lists

Collective typing - Annotate the entire column of a table or a list

<table>
<thead>
<tr>
<th>Paper</th>
<th>Taxonomy</th>
<th>Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Limaye et al. 10]</td>
<td>YAGO</td>
<td>Conditional random field for joint annotation (supervised)</td>
</tr>
</tbody>
</table>
| [Venetis et al. 11] | Google’s IsA DB | 1) Max likelihood (hit entities between 10% and 50%) \( \prod_e p(e|c) \)  
\[ p(e|c) \propto p(c|e)/p(c) \] is smoothed by frequency of (e,c) in DB;  
2) Majority (>50%) voting \( \sum_e \frac{1}{\text{Rank}(c|e)} \) |
| [Song et al. 11] | MS.KB | Naïve Bayes \( p(c) \prod_e p(e|c) \)  
\( p(e|c) \) is smoothed by frequency of (e,c) in MS.KB |
| [Pimplikar & Sarawagi 12] | Ad-hoc column name | A dual problem: find columns that best fit query concepts  
Conditional random fields (supervised) |
Outline

1. Introduction to mining latent entity structures
2. Mining latent topic hierarchies
3. Mining latent entity relations
4. Mining latent entity concepts
5. Trends and research problems
Open Problems on Mining Latent Entity Structures

What is the best way to organize information and interact with users?
Understand the Data

Coverage & Volatility

- System, architecture and database
  - How do we design such a multi-layer organization system?

- Information quality and security
  - How do we control information quality and resolve conflicts?

Diagram:
- Data record
  - Papers, tweets...
- Dataset
  - Archives, samples....
- Information source
  - NYT, twitter...
- Knowledge base
  - Freebase, Satori...
Understand the People

- NLP, ML, AI: Understand & answer natural language questions
- HCI, Crowdsourcing, Web search, domain experts: Explore latent structures with user guidance
Mining Relations & Concepts from Multiple Sources

- Knowledgebase
- Taxonomy
- Web tables
- Web pages
- Domain text
- Social media
- Social networks

...
Integration of NLP & Data Mining Approaches

NLP - analyzing single sentences

Data mining - analyzing big data

Source  Response  System
<Claim, Evidence>
Enhance Individual SF Systems

Truth finding efficiency

Application to the slot filling task [Yu et al. 14b]
Mining Latent Entity Structures from Massive Unstructured & Linked Data

Concepts

Integrate multiple sources

Frequent pattern mining + probabilistic analysis

Relations

Integrate NLP & data mining approaches

entity

text

Topics
Biography

Jiawei Han ([http://www.cs.uiuc.edu/hanj](http://www.cs.uiuc.edu/hanj)), Abel Bliss Professor, Computer Science, UIUC. He has been researching into data mining, database systems, and information networks, with over 600 conference and journal publications. He is Fellow of ACM and Fellow of IEEE, the first author of the textbook “Data Mining: Concepts and Techniques" 3rd ed., (Morgan Kaufmann, 2011), and the Director of Information Network Academic Research Center, supported by the U.S. Army Research Lab since 2009. He has delivered over 20 tutorials in major conferences including SIGMOD, VLDB, KDD, WWW, WSDM, ICDE, ICDM, SDM, ASONAM, and CIKM.

Chi Wang ([http://cs.uiuc.edu/homes/chiwang1](http://cs.uiuc.edu/homes/chiwang1)), Ph.D. candidate in Computer Science, UIUC. His Ph.D. research has been focused on mining hidden entity structures from massive data and information networks, with over 30 research publications in refereed journals and conferences including KDD, WWW, SIGIR, ICDM, SDM, CIKM, ECMLPKDD, IJCAI, and SIGMOD. He is the winner of the prestige Microsoft Research Graduate Research Fellowship, KDDCUP’13 runner-up award, and best paper award candidates in CIKM’13 and ICDM’13.
References


6. [Li et al. 13] Y. Li, C. Wang, F. Han, J. Han, D. Roth, and X. Yan. Mining Evidences for Named Entity Disambiguation, KDD’13.
References


References

12. [Sun et al. 11] Y. Sun, J. Han, X. Yan, P. S. Yu, T. Wu. Pathsim: Meta path-based top-k similarity search in heterogeneous information networks, VLDB’11.


References


25. [Li & McCallum 06] W. Li, A. McCallum. Pachinko allocation: Dag-structured mixture models of topic correlations, ICML’06.


References


References

36. [Zhao et al. 11] W. X. Zhao, J. Jiang, J. He, Y. Song, P. Achananuparp, E.-P. Lim, X. Li. Topical keyphrase extraction from twitter, HLT’11.


38. [Sun et al. 09a] Y. Sun, J. Han, J. Gao, Y. Yu. itopicmodel: Information network-integrated topic modeling, ICDM’09.

39. [Deng et al. 11] H. Deng, J. Han, B. Zhao, Y. Yu, C. X. Lin. Probabilistic topic models with biased propagation on heterogeneous information networks, KDD’11.


References

42. [Kim et al. 12c] H. Kim, Y. Sun, J. Hockenmaier, J. Han. Etm: Entity topic models for mining documents associated with entities, ICDM’12.

43. [Cohn & Hofmann 01] D. Cohn, T. Hofmann. The missing link-a probabilistic model of document content and hypertext connectivity, NIPS’01.


46. [Sun et al. 09b] Y. Sun, Y. Yu, J. Han. Ranking-based clustering of heterogeneous information networks with star network schema, KDD’09.

References


52. [McCallum et al. 05] A. McCallum, A. Corrada-Emmanuel, X. Wang. Topic and role discovery in social networks, IJCAI’05.

References


References

60. [Cucerzan 07] S. Cucerzan. Large-scale named entity disambiguation based on wikipedia data, EMNLP-CoNLL’07.

61. [Ratinov et al. 11] L. Ratinov, D. Roth, D. Downey, M. Anderson. Local and global algorithms for disambiguation to wikipedia, ACL’11.


