Bringing Structure to Text

Jiawei Han, Chi Wang and Ahmed El-Kishky
Computer Science, University of Illinois at Urbana-Champaign
August 24, 2014
Outline

1. Introduction to bringing structure to text
2. Mining phrase-based and entity-enriched topical hierarchies
3. Heterogeneous information network construction and mining
4. Trends and research problems
Motivation of Bringing Structure to Text

- 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 of 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
- Extracted 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 #maythefourthbewithyou

The White House
Happy Star Wars Day! Not building a Death Star.

URLs:
- flic.kr/p/75XWNY
- wh.gov/Pttj
Text-Attached Information Network for Unstructured and Loosely-Structured Data

- venue
- location
- organization
- venue
- location
- organization
- hashtag
- twitter
- URL
- text
- entity (given or extracted)
- author
- papers
- news
- person
- tweets
- text
- entity (given or extracted)
- author
- papers
- news
- person
- tweets
- text
- entity (given or extracted)
What Power Can We Gain if More Structures Can Be Discovered?

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

**Christos Faloutsos**
Carnegie Mellon University, 2000 – 2010
University of Maryland, 1986 – 1997

**Author Rankings**
Temporal and Spatial Databases: 190
Database and Information System: 9
Data Mining: 4

**Frequent co-authors**
- Sprios Papadimitriou
- Jinieng Sun
- Aghna J. M. Traina
- Hanghang Tong
- Jia-Yu Pan

**Advisees**
<table>
<thead>
<tr>
<th>Name</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?
Democrats ask: can health care bill be saved? Washington president barack obama’s pledge to keep fighting for major health care legislation on thursday seemed altogether detached from the legis...
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

- DBLP Bibliographic Network
- The IMDB Movie Network
- The Facebook Network
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>
Useful Structure from Text: Phrases, Topics, Entities

- Top 10 active politicians and phrases regarding healthcare issues?
- Top 10 researchers and phrases in data mining and their specializations?
Outline

1. Introduction to bringing structure to text
2. Mining phrase-based and entity-enriched topical hierarchies
3. Heterogeneous information network construction and mining
4. Trends and research problems
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 -> phrases
   iii) Text -> text + entity

C. An integrated framework
Methodologies of Topic Mining

A. Traditional bag-of-words topic modeling

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

C. An integrated framework
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

Generative process with unknown parameters $\Theta$

Corpus

- Criticism of government response to the hurricane ...

Two representative models: pLSA and LDA

- Model Inference: Fit the model with observed documents to recover the unknown parameters
Probabilistic Latent Semantic Analysis (PLSA) [Hofmann 99]

- \( k \) topics: \( k \) multinomial distributions over words
- \( D \) documents: \( D \) multinomial 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 $\theta_d$ (e.g. $z=1$)
2. Sample a word $w$ according to $\phi_z$ (e.g. $w=$government)
PLSA – Model Inference using Expectation-Maximization (EM)

Topic $\phi_1$
- government ?
- response ?

... Topic $\phi_k$
- donate ?
- relief ?

Corpus

Doc $\theta_1$
- ?
- ?
- ?

... Doc $\theta_D$
- ?
- ?
- ?

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

Bayes rule

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

**PROS**
- Simple, only one hyperparameter k
- Easy to incorporate prior in the EM algorithm

**CONS**
- High model complexity -> prone to overfitting
- The EM solution is neither optimal nor unique
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 $\phi, \theta$

- Same as pLSA

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 $\mathbf{z}_{-i}$

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

- **Topic $\phi_1$**
  - government?
  - response?

- **Topic $\phi_k$**
  - donate?
  - relief?

- **corpus**
  - Criticism of government response to the hurricane...

- What parameters fit the empirical moments?

- What parameters are most likely to generate the observed corpus?

**Moments: expectation of patterns**

<table>
<thead>
<tr>
<th></th>
<th>criticism</th>
<th>response</th>
<th>government</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>length 1</strong></td>
<td>0.03</td>
<td>0.01</td>
<td>0.04</td>
</tr>
<tr>
<td><strong>length 2</strong> (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>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>criticism government response</td>
<td>0.001</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Guaranteed Topic Recovery

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

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

- **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

V: vocabulary size; k: topic number
Tensor Orthogonal Decomposition for LDA

Input corpus

Normalized pattern counts

<table>
<thead>
<tr>
<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>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>
</tbody>
</table>

V: vocabulary size
k: topic number

V_2

V_3

eigen decomposition

tensor product

Topic \( \phi_1 \)

government 0.3
response 0.2
...

Topic \( \phi_k \)

donate 0.1
relief 0.05
...

[ANANDKUMAR ET AL. 12] 35
Tensor Orthogonal Decomposition for LDA – Not Scalable

Input corpus

Normalized pattern counts

<table>
<thead>
<tr>
<th></th>
<th>A: 0.03</th>
<th>AB: 0.001</th>
<th>ABC: 0.001</th>
<th>B: 0.01</th>
<th>BC: 0.002</th>
<th>ABD: 0.005</th>
<th>C: 0.04</th>
<th>AC: 0.003</th>
<th>BCD: 0.004</th>
</tr>
</thead>
</table>
| V: vocabulary size; k: topic number | L: # tokens; l: average doc length

Normalized pattern counts

Prohibitive to compute

Time: $O(V^3 k + Ll^2)$
Space: $O(V^3)$
Scalable Tensor Orthogonal Decomposition

<table>
<thead>
<tr>
<th>Normalized pattern counts</th>
</tr>
</thead>
<tbody>
<tr>
<td>A: 0.03</td>
</tr>
<tr>
<td>AB: 0.001</td>
</tr>
<tr>
<td>ABC: 0.001</td>
</tr>
<tr>
<td>B: 0.01</td>
</tr>
<tr>
<td>BC: 0.002</td>
</tr>
<tr>
<td>ABD: 0.005</td>
</tr>
<tr>
<td>C: 0.04</td>
</tr>
<tr>
<td>AC: 0.003</td>
</tr>
<tr>
<td>BCD: 0.004</td>
</tr>
<tr>
<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]
Speedup 1
Eigen-Decomposition of $M_2$

$$M_2 = E_2 - c_1 E_1 \otimes E_1 \in \mathbb{R}^{V^*V}$$

1. Eigen-decomposition of $E_2$

$$\Rightarrow (M_2 = U_1\tilde{M}_2 U_1^T)$$

$$\Sigma_1 - c_1 (U_1^T E_1) \otimes (U_1^T E_1)$$

| AB: 0.001 |
| BC: 0.002 |
| AC: 0.003 |
| :         |

$E_2$ (Sparse) $\leftrightarrow V$

$U_1$(Eigenvec) $\leftrightarrow V$

$\Sigma_1$ $\leftrightarrow k$

$U_1^T$ $\leftrightarrow V$
Speedup 1
Eigen-Decomposition of $M_2$

$M_2 = (U_1U_2)\Sigma(U_1U_2)^T = \Sigma\Sigma^T$

1. Eigen-decomposition of $E_2$
   $\Rightarrow (M_2 = U_1\tilde{M}_2U_1^T)$

2. Eigen-decomposition of $\tilde{M}_2$
Speedup 2
Construction of Small Tensor

\[ \tilde{T} = M_3(W, W, W) \]

\[ W = \Sigma^{-\frac{1}{2}}, W^T M_2 W = I \]

\[ (v \otimes^3)(W, W, W) = (W^T v) \otimes^3 \]

\[ (v \otimes E_2)(W, W, W) = W^T v \otimes W^T E_2 W \]
20-3000 Times Faster

- Two scans vs. thousands of scans

STOD – Scalable tensor orthogonal decomposition
TOD – Tensor orthogonal decomposition
Gibbs Sampling – Collapsed Gibbs sampling

Synthetic data
Real data

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Loaded into Memory</th>
<th>Not Loaded into Memory</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$L=19M$</td>
<td>$L=39M$</td>
</tr>
<tr>
<td>news</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STOD</td>
<td>293</td>
<td>310</td>
</tr>
<tr>
<td>TOD</td>
<td>6877</td>
<td>768110</td>
</tr>
<tr>
<td>Gibbs sampling</td>
<td>21641</td>
<td>48999</td>
</tr>
</tbody>
</table>

| CS      |                    |                        |
| STOD    | 541                | 577                    |
| TOD     | 14439              | 1661101                |
| Gibbs sampling | 47293          | 102136                 |
Effectiveness

**STOD = TOD > Gibbs Sampling**

- Recovery error is low when the sample is large enough
- Variance is almost 0
- Coherence is high
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 -> phrases
- iii) Text -> text + entity

C. An integrated framework
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 $\phi_z$.

```
Topic $\phi_1$

government 0.3
response 0.2
...

Topic $\phi_k$

donate 0.1
relief 0.05
...
```

```
Information technology & system
IR
o/1/1  o/1/2
o/2/1  o/2/2

CS
o

DB
```

$$\text{Information technology & system}$$
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
LDA with Topic Tree

Latent Dirichlet Allocation with Topic Tree

$\alpha$

Dirichlet prior

$\theta$

$z_1 \rightarrow \ldots \rightarrow z_n$

$\phi$

Word distributions

$\alpha_0$

$\alpha_{o/1}$

$\phi_{o/1/1}$

$\phi_{o/1/2}$

$\alpha_{o/2}$

$\Phi_o$

$\alpha_{o/2/1}$

$\alpha_{o/2/2}$

$\phi_{o/2/1}$

$\phi_{o/2/2}$

$\#\text{words in } d$

$\#\text{docs}$

$\text{Topic distributions}$

$\text{Word distributions}$

[WANG ET AL. 14B]
Recursive Inference for LDA with Topic Tree

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

Flexible to decide when to terminate

Easy to revise the tree structure

Inference order

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

Theorem. STROD ensures robust recovery and revision
Methodologies of Topic Mining

A. Traditional bag-of-words topic modeling

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

C. An integrated framework
Unigrams -> N-Grams

- Motivation: unigrams can be difficult to interpret

The topic that represents the area of Machine Learning

<table>
<thead>
<tr>
<th>Unigrams</th>
<th>N-Grams</th>
</tr>
</thead>
<tbody>
<tr>
<td>learning</td>
<td>learning</td>
</tr>
<tr>
<td>reinforcement</td>
<td>support vector machines</td>
</tr>
<tr>
<td>support</td>
<td>reinforcement learning</td>
</tr>
<tr>
<td>machine</td>
<td>feature selection</td>
</tr>
<tr>
<td>vector</td>
<td>conditional random fields</td>
</tr>
<tr>
<td>selection</td>
<td>classification</td>
</tr>
<tr>
<td>feature</td>
<td>decision trees</td>
</tr>
<tr>
<td>random</td>
<td>:</td>
</tr>
<tr>
<td>:</td>
<td>:</td>
</tr>
</tbody>
</table>

versus

The topic that represents the area of Machine Learning
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 – Simultaneously Inferring Phrases and Topic

- **Bigram Topic Model** ([Wallach 06]) – probabilistic generative model that conditions on previous word and topic when drawing next word.

- **Topical N-Grams** ([Wang et al. 07]) – probabilistic model that generates words in textual order. Creates n-grams by concatenating successive bigrams (Generalization of Bigram Topic Model).

- **Phrase-Discovering LDA (PDLDA)** ([Lindsey et al. 12]) – Viewing each sentence as a time-series of words, PDLDA posits that the generative parameter (topic) changes periodically. Each word is drawn based on previous m words (context) and current phrase topic.
Strategy 1 – Bigram Topic Model

To generate a token in document:

1. Sample a topic label $z$ according to $\theta_d$
2. Sample a word $w$ according to $z$ and the previous token

- Overall quality of inferred topics is improved by considering bigram statistics and word order
- Interpretability of bigrams is not considered

Better quality topic model  All consecutive bigrams generated  Fast inference

[WALLACH ET AL. 06]
Strategy 1 – Topical N-Grams Model (TNG)

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
Words in phrase do not share topic
High inference cost - slow

[WANG ET AL. 07, LINDSEY ET AL. 12]
<table>
<thead>
<tr>
<th>LDA</th>
<th>Reinforcement Learning</th>
<th>Human Receptive System</th>
</tr>
</thead>
<tbody>
<tr>
<td>state</td>
<td>reinforcement learning</td>
<td>motion</td>
</tr>
<tr>
<td>learning</td>
<td>optimal policy</td>
<td>visual</td>
</tr>
<tr>
<td>policy</td>
<td>dynamic programming</td>
<td>spatial frequency</td>
</tr>
<tr>
<td>action</td>
<td>optimal control</td>
<td>reinforcement</td>
</tr>
<tr>
<td>reinforcement</td>
<td>function approximator</td>
<td>states</td>
</tr>
<tr>
<td>states</td>
<td>prioritized sweeping</td>
<td>field</td>
</tr>
<tr>
<td>time</td>
<td>finite-state controller</td>
<td>position</td>
</tr>
<tr>
<td>optimal actions</td>
<td>learning system</td>
<td>figure</td>
</tr>
<tr>
<td>function</td>
<td>q-learning</td>
<td>direction</td>
</tr>
<tr>
<td>algorithm</td>
<td>markov decision problems</td>
<td>fields</td>
</tr>
<tr>
<td>reward</td>
<td>markov decision processes</td>
<td>eye</td>
</tr>
<tr>
<td>step</td>
<td>local search</td>
<td>location</td>
</tr>
<tr>
<td>dynamic control</td>
<td>state-action pair</td>
<td>preferred direction</td>
</tr>
<tr>
<td>sutton</td>
<td>belief states</td>
<td>visual processing</td>
</tr>
<tr>
<td>rl</td>
<td>stochastic policy</td>
<td>velocity</td>
</tr>
<tr>
<td>decision</td>
<td>action selection</td>
<td>receptive</td>
</tr>
<tr>
<td>algorithms</td>
<td>upright position</td>
<td>velocity</td>
</tr>
<tr>
<td>agent</td>
<td>reinforcement learning methods</td>
<td>light</td>
</tr>
<tr>
<td>LDA</td>
<td>Speech Recognition</td>
<td>LDA</td>
</tr>
<tr>
<td>-------------</td>
<td>-----------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>recognition</td>
<td>speech recognition</td>
<td>kernel</td>
</tr>
<tr>
<td>system</td>
<td>training data</td>
<td>linear</td>
</tr>
<tr>
<td>word</td>
<td>neural network</td>
<td>vector</td>
</tr>
<tr>
<td>face</td>
<td>error rates</td>
<td>support</td>
</tr>
<tr>
<td>context</td>
<td>neural net</td>
<td>set</td>
</tr>
<tr>
<td>character</td>
<td>hidden markov model</td>
<td>nonlinear</td>
</tr>
<tr>
<td>hmm</td>
<td>feature vectors</td>
<td>data</td>
</tr>
<tr>
<td>based</td>
<td>continuous speech</td>
<td>algorithm</td>
</tr>
<tr>
<td>frame</td>
<td>training procedure</td>
<td>space</td>
</tr>
<tr>
<td>segmentation</td>
<td>continuous speech recognition</td>
<td>pca</td>
</tr>
<tr>
<td>training</td>
<td>gamma filter</td>
<td>function</td>
</tr>
<tr>
<td>characters</td>
<td>hidden control</td>
<td>problem</td>
</tr>
<tr>
<td>set</td>
<td>speech production</td>
<td>margin</td>
</tr>
<tr>
<td>probabilities</td>
<td>neural nets</td>
<td>vectors</td>
</tr>
<tr>
<td>features</td>
<td>input representation</td>
<td>solution</td>
</tr>
<tr>
<td>faces</td>
<td>output layers</td>
<td>training</td>
</tr>
<tr>
<td>words</td>
<td>training algorithm</td>
<td>points</td>
</tr>
<tr>
<td>frames</td>
<td>test set</td>
<td></td>
</tr>
<tr>
<td>database</td>
<td>speech frames</td>
<td></td>
</tr>
<tr>
<td>mlp</td>
<td>speaker dependent</td>
<td></td>
</tr>
</tbody>
</table>
Strategy 1 – Phrase Discovering Latent Dirichlet Allocation

To generate a token in a document:
- Let \( u \), a context vector consisting of the shared phrase topic and the past \( m \) words.
- Draw a token from the Pitman-Yor Process conditioned on \( u \)

When \( m = 1 \), this generative model is equivalent to TNG

High model complexity - overfitting
Principled topic assignment
High inference cost - slow

[WANG ET AL. 07, LINDSEY ET AL. 12]
<table>
<thead>
<tr>
<th>words</th>
<th>main idea</th>
<th>word processing center</th>
</tr>
</thead>
<tbody>
<tr>
<td>word</td>
<td>topic sentence</td>
<td>word processing systems</td>
</tr>
<tr>
<td>sentence</td>
<td>english language</td>
<td>word processing equipment</td>
</tr>
<tr>
<td>write</td>
<td>following paragraph</td>
<td>speak different languages</td>
</tr>
<tr>
<td>writing</td>
<td>words like</td>
<td>use quotation marks</td>
</tr>
<tr>
<td>paragraph</td>
<td>quotation marks</td>
<td>single main idea</td>
</tr>
<tr>
<td>sentences</td>
<td>direct object</td>
<td>use words like</td>
</tr>
<tr>
<td>meaning</td>
<td>word processing</td>
<td>topic sentence states</td>
</tr>
<tr>
<td>use</td>
<td>sentence tells</td>
<td>present perfect tense</td>
</tr>
<tr>
<td>subject</td>
<td>figurative language</td>
<td>express complete thoughts</td>
</tr>
<tr>
<td>language</td>
<td>writing process</td>
<td>word processing software</td>
</tr>
<tr>
<td>read</td>
<td>following sentences</td>
<td>use formal english</td>
</tr>
<tr>
<td>example</td>
<td>subject matter</td>
<td>standard american english</td>
</tr>
<tr>
<td>verb</td>
<td>standard english</td>
<td>collective noun refers</td>
</tr>
<tr>
<td>topic</td>
<td>use words</td>
<td>formal standard english</td>
</tr>
</tbody>
</table>

(c) Topic 3

| energy     | natural resources                            | nuclear power plants                    |
| used       | natural gas                                   | nuclear power plant                     |
| oil        | heat energy                                   | important natural resources             |
| heat       | iron ore                                      | electric power plants                   |
| coal       | carbon dioxide                                | called fossil fuels                     |
| use        | potential energy                              | important natural resource              |
| fuel       | solar energy                                  | produce large amounts                   |
| produce    | light energy                                  | called solar energy                     |
| power      | fossil fuels                                  | electric light bulb                     |
| source     | hot water                                     | use electrical energy                   |
| light      | steam engine                                  | use solar energy                        |
| electricity| large amounts                                 | carbon dioxide gas                      |
| burn       | sun's energy                                  | called potential energy                 |
| gas        | radiant energy                                 | gas called carbon dioxide               |
| gasoline   | nuclear energy                                | called crude oil                        |

(d) Topic 4

PD-LDA: Experiments on the Touchstone Applied Science Associates (TASA) corpus
<table>
<thead>
<tr>
<th>matter</th>
<th>chemical reactions</th>
<th>like charges repel</th>
<th>positively charged nucleus</th>
</tr>
</thead>
<tbody>
<tr>
<td>atoms</td>
<td>atomic number</td>
<td>unlike charges attract</td>
<td>outer energy level</td>
</tr>
<tr>
<td>elements</td>
<td>hydrogen atoms</td>
<td>reaction takes place</td>
<td>negatively charged electrons</td>
</tr>
<tr>
<td>electrons</td>
<td>hydrogen atom</td>
<td>chemical change takes place</td>
<td>chemical change takes place</td>
</tr>
<tr>
<td>atom</td>
<td>periodic table</td>
<td>form new substances</td>
<td>form new substances</td>
</tr>
<tr>
<td>molecules</td>
<td>chemical change</td>
<td>physical properties</td>
<td>physical change takes place</td>
</tr>
<tr>
<td>form</td>
<td>chemical reaction</td>
<td>water molecules</td>
<td>form sodium chloride</td>
</tr>
<tr>
<td>oxygen</td>
<td>chemical reaction</td>
<td>sodium chloride</td>
<td>modern atomic theory</td>
</tr>
<tr>
<td>hydrogen</td>
<td>water molecules</td>
<td>physical change takes place</td>
<td>electrically charged particles</td>
</tr>
<tr>
<td>particles</td>
<td>sodium chloride</td>
<td>form sodium chloride</td>
<td>increasing atomic number</td>
</tr>
<tr>
<td>element</td>
<td>small amounts</td>
<td>modern atomic theory</td>
<td>second ionization energies</td>
</tr>
<tr>
<td>solution</td>
<td>positive charge</td>
<td>electrically charged particles</td>
<td>higher energy levels</td>
</tr>
<tr>
<td>substance</td>
<td>carbon atoms</td>
<td>increasing atomic number</td>
<td>second ionization energies</td>
</tr>
<tr>
<td>reaction</td>
<td>physical change</td>
<td>second ionization energies</td>
<td>higher energy levels</td>
</tr>
<tr>
<td>nucleus</td>
<td>chemical properties</td>
<td>increasing atomic number</td>
<td>second ionization energies</td>
</tr>
</tbody>
</table>

**(b) Topic 2**

- president
- supreme court
- congress
- new york
- vote
- democratic party
- party
- vice president
- constitution
- political parties
- state
- national government
- members
- executive branch
- office
- civil rights
- government
- new government
- states
- political party
- elected
- andrew jackson
- representatives
- chief justice
- senate
- federal government
- house
- state legislatures
- washington
- public opinion

**PD-LDA: Experiments on the Touchstone Applied Science Associates (TASA) corpus**
Strategy 2 – Post topic modeling phrase construction

- TurboTopics [Blei & Lafferty 09] – Phrase construction as a post-processing step to Latent Dirichlet Allocation
  - Merges adjacent unigrams with same topic label if merge significant.

- KERT [Danilevsky et al] – Phrase construction as a post-processing step to Latent Dirichlet Allocation
  - Performs frequent pattern mining on each topic
  - Performs phrase ranking on four different criterion
Strategy 2 – TurboTopics

Annotated documents

What is phase\textsubscript{11} transition\textsubscript{11}? Why is there phase\textsubscript{11} transition\textsubscript{11}? These are old\textsubscript{127} questions\textsubscript{127} people\textsubscript{170} have been asking\textsubscript{195} for many years\textsubscript{127} but get\textsubscript{153} few answers\textsubscript{127}. We established\textsubscript{127} one general\textsubscript{11} theory\textsubscript{127} based\textsubscript{153} on game\textsubscript{153} theory\textsubscript{127} and topology\textsubscript{85} it provides\textsubscript{11} a basic\textsubscript{127} understanding\textsubscript{127} to phase\textsubscript{11} transition\textsubscript{11}. We proposed\textsubscript{11} a modern\textsubscript{127} definition\textsubscript{117} of phase\textsubscript{11} transition\textsubscript{11} based\textsubscript{153} on game\textsubscript{153} theory\textsubscript{127} and topology\textsubscript{85} of symmetry\textsubscript{11} group\textsubscript{184} which unified\textsubscript{135} Ehrenests definition\textsubscript{117}. A spontaneous\textsubscript{11} result\textsubscript{68} of this topological\textsubscript{85} phase\textsubscript{11} transition\textsubscript{11} theory\textsubscript{127} is the universal\textsubscript{14} equation\textsubscript{117} of coexistence\textsubscript{195} curve\textsubscript{195} in phase\textsubscript{11} diagram\textsubscript{11} it holds\textsubscript{153} both for classical\textsubscript{122} and quantum\textsubscript{11} phase\textsubscript{11} transition\textsubscript{11}. This

LDA topic #11

phase, transitions, phases, transition, quantum, critical, symmetry, field, point, model, order, diagram, systems, two, theory, system, study, breaking, spin, first

Turbo topic #11

phase transitions, model, symmetry, point, quantum, systems, phase transition, phase diagram, system, order, field, order, parameter, critical, two, transitions in, models, different, symmetry breaking, first order, phenomena

[BLEI ET AL. 09]
Strategy 2 – TurboTopics

TurboTopics methodology:
1. Perform Latent Dirichlet Allocation on corpus to assign each token a topic label
2. For each topic find adjacent unigrams that share the same latent topic, then perform a distribution-free permutation test on arbitrary-length back-off model.

End recursive merging when all significant adjacent unigrams have been merged.
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.
...

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

Unigram topic assignment: Topic 1 & Topic 2
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 Modeling


**ToPMine framework:**
1. Perform frequent *contiguous pattern* mining to extract candidate phrases and their counts
2. Perform agglomerative merging of adjacent unigrams as guided by a significance score. This segments each document into a “bag-of-phrases”
3. The newly formed bag-of-phrases are passed as input to PhraseLDA, an extension of LDA that constrains all words in a phrase to each share the same latent topic.
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

More challenging than in strategy 2!

Topic model inference with phrase constraints

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

Significance score

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

\[ (Markov Blanket) \quad (Feature Selection) \quad (for) \quad (Support Vector Machines) \]

\[ \alpha = 0 \quad \alpha = 0 \quad \alpha = 0 \]

\[ \alpha = 5 \text{ threshold} \]

Markov Blanket Feature Selection for Support Vector Machines.

Good Phrases
Phrase Mining: Frequent Pattern Mining + Statistical Analysis

Significance score

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

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

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

…[support vector] for [machine learning]…
Collocation Mining

- A collocation is a sequence of words that occur more frequently than is expected. These collocations can often be quite “interesting” and due to their non-compositionality, often relay information not portrayed by their constituent terms (e.g., “made an exception”, “strong tea”)

- There are many different measures used to extract collocations from a corpus [Ted Dunning 93, Ted Pederson 96]
  - mutual information, t-test, z-test, chi-squared test, likelihood ratio

- Many of these measures can be used to guide the agglomerative phrase-segmentation algorithm
ToPMine: Phrase LDA (Constrained Topic Modeling)

- Generative model for PhraseLDA is the same as LDA.
- The model incorporates constraints obtained from the “bag-of-phrases” input.
- Chain-graph shows that all words in a phrase are constrained to take on the same topic values.

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

Topic model inference with phrase constraints
Example Topical Phrases

### PDLDA [Lindsey et al. 12] – Strategy 1 (3.72 hours)

<table>
<thead>
<tr>
<th>social networks</th>
<th>information retrieval</th>
</tr>
</thead>
<tbody>
<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>
<tr>
<td>:</td>
<td>:</td>
</tr>
<tr>
<td>Topic 1</td>
<td>Topic 2</td>
</tr>
</tbody>
</table>

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

<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>
<tr>
<td>:</td>
<td>:</td>
</tr>
<tr>
<td>Topic 1</td>
<td>Topic 2</td>
</tr>
<tr>
<td>Topic 1</td>
<td>Topic 2</td>
</tr>
<tr>
<td>-------------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>unigrams</td>
<td></td>
</tr>
<tr>
<td>problem</td>
<td>word</td>
</tr>
<tr>
<td>algorithm</td>
<td>language</td>
</tr>
<tr>
<td>optimal</td>
<td>text</td>
</tr>
<tr>
<td>solution</td>
<td>speech</td>
</tr>
<tr>
<td>search</td>
<td>system</td>
</tr>
<tr>
<td>solve</td>
<td>recognition</td>
</tr>
<tr>
<td>constraints</td>
<td>character</td>
</tr>
<tr>
<td>programming</td>
<td>translation</td>
</tr>
<tr>
<td>heuristic</td>
<td>sentences</td>
</tr>
<tr>
<td>genetic</td>
<td>grammar</td>
</tr>
<tr>
<td>n-grams</td>
<td></td>
</tr>
<tr>
<td>genetic algorithm</td>
<td>natural language</td>
</tr>
<tr>
<td>optimization problem</td>
<td>speech recognition</td>
</tr>
<tr>
<td>solve this problem</td>
<td>language model</td>
</tr>
<tr>
<td>optimal solution</td>
<td>natural language</td>
</tr>
<tr>
<td>evolutionary algorithm</td>
<td>translation</td>
</tr>
<tr>
<td>local search</td>
<td>recognition system</td>
</tr>
<tr>
<td>search space</td>
<td>context free grammars</td>
</tr>
<tr>
<td>optimization algorithm</td>
<td>sign language</td>
</tr>
<tr>
<td>search algorithm</td>
<td>recognition rate</td>
</tr>
<tr>
<td>objective function</td>
<td>character recognition</td>
</tr>
<tr>
<td>Topic 1</td>
<td>Topic 2</td>
</tr>
<tr>
<td>-----------------</td>
<td>------------------</td>
</tr>
<tr>
<td>unigrams</td>
<td></td>
</tr>
<tr>
<td>plant</td>
<td>church</td>
</tr>
<tr>
<td>nuclear</td>
<td>catholic</td>
</tr>
<tr>
<td>environmental</td>
<td>religious</td>
</tr>
<tr>
<td>energy</td>
<td>bishop</td>
</tr>
<tr>
<td>year</td>
<td>pope</td>
</tr>
<tr>
<td>waste</td>
<td>roman</td>
</tr>
<tr>
<td>department</td>
<td>jewish</td>
</tr>
<tr>
<td>power</td>
<td>rev</td>
</tr>
<tr>
<td>state</td>
<td>john</td>
</tr>
<tr>
<td>chemical</td>
<td>christian</td>
</tr>
<tr>
<td>n-grams</td>
<td></td>
</tr>
<tr>
<td>energy department</td>
<td>roman catholic</td>
</tr>
<tr>
<td>environmental protection agency</td>
<td>pope john paul</td>
</tr>
<tr>
<td>nuclear weapons</td>
<td>john paul</td>
</tr>
<tr>
<td>acid rain</td>
<td>catholic church</td>
</tr>
<tr>
<td>nuclear power plant</td>
<td>anti semitism</td>
</tr>
<tr>
<td>hazardous waste</td>
<td>baptist church</td>
</tr>
<tr>
<td>savannah river</td>
<td>united states</td>
</tr>
<tr>
<td>rocky flats</td>
<td>lutheran church</td>
</tr>
<tr>
<td>nuclear power</td>
<td>episcopal church</td>
</tr>
<tr>
<td>natural gas</td>
<td>church members</td>
</tr>
<tr>
<td>Topic 1</td>
<td>Topic 2</td>
</tr>
<tr>
<td>-----------------</td>
<td>--------------------------</td>
</tr>
<tr>
<td>unigrams</td>
<td></td>
</tr>
<tr>
<td>coffee</td>
<td>food</td>
</tr>
<tr>
<td>ice</td>
<td>good</td>
</tr>
<tr>
<td>cream</td>
<td>place</td>
</tr>
<tr>
<td>flavor</td>
<td>ordered</td>
</tr>
<tr>
<td>egg</td>
<td>chicken</td>
</tr>
<tr>
<td>chocolate</td>
<td>roll</td>
</tr>
<tr>
<td>breakfast</td>
<td>sushi</td>
</tr>
<tr>
<td>tea</td>
<td>restaurant</td>
</tr>
<tr>
<td>cake</td>
<td>dish</td>
</tr>
<tr>
<td>sweet</td>
<td>rice</td>
</tr>
<tr>
<td>n-grams</td>
<td></td>
</tr>
<tr>
<td>ice cream</td>
<td>spring rolls</td>
</tr>
<tr>
<td>iced tea</td>
<td>food was good</td>
</tr>
<tr>
<td>french toast</td>
<td>fried rice</td>
</tr>
<tr>
<td>hash browns</td>
<td>egg rolls</td>
</tr>
<tr>
<td>frozen yogurt</td>
<td>chinese food</td>
</tr>
<tr>
<td>eggs benedict</td>
<td>pad thai</td>
</tr>
<tr>
<td>peanut butter</td>
<td>dim sum</td>
</tr>
<tr>
<td>cup of coffee</td>
<td>thai food</td>
</tr>
<tr>
<td>iced coffee</td>
<td>pretty good</td>
</tr>
<tr>
<td>scrambled eggs</td>
<td>lunch specials</td>
</tr>
</tbody>
</table>
## Comparison of Strategies on Runtime

### Runtime evaluation

strategy 3 > strategy 2 > strategy 1

### Comparison of three strategies

<table>
<thead>
<tr>
<th>Method</th>
<th>sampled dblp titles ($k=5$)</th>
<th>dblp titles ($k=30$)</th>
<th>sampled dblp abstracts</th>
<th>dblp abstracts</th>
</tr>
</thead>
<tbody>
<tr>
<td>PDLDA Turbo Topics</td>
<td>3.72 (hrs)</td>
<td>~20.44 (days)</td>
<td>1.12 (days)</td>
<td>~95.9 (days)</td>
</tr>
<tr>
<td>TNG</td>
<td>6.68 (hrs)</td>
<td>&gt;30 (days)*</td>
<td>&gt;10 (days)*</td>
<td>&gt;50 (days)*</td>
</tr>
<tr>
<td>LDA</td>
<td>146 (s)</td>
<td>5.57 (hrs)</td>
<td>853 (s)</td>
<td>NA†</td>
</tr>
<tr>
<td>KERT</td>
<td><strong>65 (s)</strong></td>
<td>3.04 (hrs)</td>
<td>353 (s)</td>
<td>13.84 (hours)</td>
</tr>
<tr>
<td>ToP-Mine</td>
<td>68 (s)</td>
<td>3.08 (hrs)</td>
<td>1215 (s)</td>
<td>NA†</td>
</tr>
<tr>
<td></td>
<td><strong>67 (s)</strong></td>
<td><strong>2.45 (hrs)</strong></td>
<td><strong>340 (s)</strong></td>
<td><strong>10.88 (hrs)</strong></td>
</tr>
</tbody>
</table>
Comparison of Strategies on Topical Coherence

strategy 3 > strategy 2 > strategy 1
Comparison of Strategies with Phrase Intrusion

Comparison of three strategies

Phrase intrusion

strategy 3 > strategy 2 > strategy 1
Comparison of Strategies on Phrase Quality

strategy 3 > strategy 2 > strategy 1

Comparison of three strategies
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
  - generally high-quality topics and phrases

- **Strategy 3**: prior bag-of-words model inference, mine phrases and impose to the bag-of-words model
  - topic inference relies on correct segmentation of documents, but not sensitive
  - can be fast
  - generally high-quality topics and phrases
Methodologies of Topic Mining

A. Traditional bag-of-words topic modeling

B. Extension of topic modeling

i) Flat -> hierarchical

ii) Unigrams -> phrases

iii) Text -> text + entity

C. An integrated framework
Text Only -> Text + Entity

Text-only corpus

Criticism of government response to the hurricane ...

Topic $\phi_1$
- government 0.3
- response 0.2
- ...

Topic $\phi_k$
- donate 0.1
- relief 0.05
- ...

Doc $\theta_1$[0.4, 0.3, 0.3]
- ...

Doc $\theta_D$[0.2, 0.5, 0.3]

- 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
  
  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
  ...

\[ \text{SIGMOD} \]
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

\[ \theta_2 \] should be similar to \( \theta_1, \theta_3 \)

\[ \theta_1^d \] 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
    - \( \text{Surajit Chaudhuri} \): \(0.3, 0.4, 0.3\)
    - \( \text{SIGMOD} \): \(0.2, 0.5, 0.3\)

- **RESEMBLE ENTITIES TO WORDS**
  - A topic has a multinomial distribution over each type of entities
    - \( \text{Topic 1} \):
      - \( \text{KDD} \): 0.3
      - \( \text{ICDM} \): 0.2
      - \( \text{Jiawei Han} \): 0.1
      - \( \text{Christos Faloustos} \): 0.05
      - Over venues
      - Over authors

- **RESEMBLE ENTITIES TO TOPICS**
  - An entity has a multinomial distribution over words
    - \( \text{SIGMOD} \):
      - \( \text{database} \): 0.3
      - \( \text{system} \): 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 \( \phi_{z,e} \)

\[
\phi_{z,e} \sim Dir(w_1 \phi_z + w_2 \phi_e)
\]
Example topics learned by ETM

On a news dataset about Japan tsunami 2011
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, ... Over venues
  - Jiawei Han 0.1, Christos Faloustos 0.05, ... Over authors

RESEMBLE ENTITIES TO TOPICS

- An entity has a multinomial distribution over words

  - SIGMOD: database 0.3, system 0.2, ...
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 $\phi_z^e$
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

---

- **Surajit Chaudhuri**: 0.3, 0.4, 0.3
- **SIGMOD**: 0.2, 0.5, 0.3

**# params = k*(E+V)**

- **KDD 0.3**, **ICDM 0.2**, **...**
  - Over venues

- **Jiawei Han 0.1**, **Christos Faloustos 0.05**, **...**
  - Over authors

**SIGMOD**

**# 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 -> phrases

iii) Text -> text + entity

C. An integrated framework
An Integrated Framework

How to choose & integrate?

- **Hierarchy**
  - Recursive
  - Non-recursive

- **Phrasal**
  - 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

- **Entity**
  - Resemble entities to documents
    - Modeling strategy 1
  - Resemble entities to topics
    - Modeling strategy 2
  - Resemble entities to words
    - Modeling strategy 3
An Integrated Framework

Hierarchy
Recursive
Non recursive

Compatible & effective

Sequence of tokens generative model
• Strategy 1

Post inference, visualize topics with n-grams
• Strategy 2

Prior model inference, mine phrases and impose to the bag-of-words model
• Strategy 3

Resemble entities to documents
• Modeling strategy 1

Resemble entities to topics
• Modeling strategy 2

Resemble entities to words
• Modeling strategy 3
Construct A Topical Hierarchy (CATHY)

- **Hierarchy + phrase + entity**

Diagram:
- **Input collection**
- **entity**
- **text**
- **Output hierarchy with phrases & entities**

Procedure:
1. **Hierarchical topic discovery with entities**
2. **Phrase mining**
3. **Rank phrases & entities per topic**
Mining Framework – CATHY

Construct A Topical Hierarchy

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

Input collection

Output hierarchy with phrases & entities

entity
text
Hierarchical Topic Discovery with Text + Multi-Typed Entities [Wang et al. 13b,14c]

Every topic has a multinomial distribution over each type of entities.

- **Topic 1**
  - \( \phi_1^1 \) data 0.2
  - \( \phi_1^2 \) Jiawei Han 0.1
  - \( \phi_1^3 \) KDD 0.3

- **Topic k**
  - \( \phi_k^1 \) database 0.2
  - \( \phi_k^2 \) Surajit Chaudhuri 0.1
  - \( \phi_k^3 \) SIGMOD 0.3

- **Authors**
  - Christos Faloustos 0.05
  - Jeff Naughton 0.05

- **Venues**
  - ICDM 0.2
  - VLDB 0.3
Text and Links: Unified as Link Patterns

Computing machinery and intelligence

A.M. Turing
Link-Weighted Heterogeneous Network

A.M. Turing

intelligence

SIGMOD

database

system

venue

text

author

word

author

venue
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 $\rho$

Suppose $t_1 = t_2 = \text{word}$
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 $\rho$
2. Sample the first end node $u$ according to $\phi_z^{t_1}$

Suppose $t_1 = t_2 = \text{word}$

*database*

*system*

*database 0.2*

*system 0.1*

*...*
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 $\rho$
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 $t_1 = t_2 = \text{word}$
Generative Model for Link Patterns - Collapsed Model

Equivalently, we can generate the number of links between $u$ and $v$:

$$e_{u,v} = e_{u,v}^1 + \cdots + e_{u,v}^k, \quad e_{u,v}^z \sim Poisson \left( \rho_z \phi_{z,u}^{t_1} \phi_{z,v}^{t_2} \right)$$

Suppose $t_1 = t_2 = word$
Theorem. The solution derived from the collapsed model

\[ e^{x,y,t}_{i,j} \sim \text{Pois}(\sum_z M^t \theta_{x,y} \rho_z \phi_{z,u}^{t_1} \phi_{z,v}^{t_2}) \]
Model Inference

**E-step. Posterior prob of latent topic for every link (Bayes rule)**

\[
e_{i,j}^{x,y,z} = \frac{e_{i,j}^{x,y,t} \rho_z \phi_i^{x,z} \phi_j^{y,z}}{\sum_{c=1}^k \rho_c \phi_i^{x,c} \phi_j^{y,c} + \rho_0 \phi_i^{x,0} \phi_j^{y,z}}
\]

**M-step. Estimate model params (Sum & normalize soft counts)**

\[
\theta_{x,y} = \frac{\sum_v e_{i,j}^{x,y,t}}{M^t}
\]

\[
\rho_z = \frac{\sum_{i,j} e_{i,j}^{x,y,z}}{\sum_{i,j,v} e_{i,j}^{x,y,z}}
\]

\[
\phi_i^{x,z} = \frac{\sum_v e_{i,j}^{x,y,z} + e_{j,i}^{y,x,z}}{\sum_{u,j,v} (e_{u,j}^{x,y,z} + e_{j,u}^{y,x,z})}
\]

\[
\phi_j^{y,z} = \frac{\sum_v e_{i,j}^{x,y,z} + e_{j,i}^{y,x,z}}{\sum_{u,j,v} (e_{u,j}^{x,y,z} + e_{j,u}^{y,x,z})}
\]

\[
e_{i,j}^{x,y,t} \sim \text{Pois}(\sum_z M^t \theta_{x,y} \rho_z \phi_i^{t_1} \phi_j^{t_2})
\]
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 scaleup of all the link weights

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

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

- **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>

Coherence of each topic - average pointwise mutual information (PMI)
Phrase Mining

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

Output hierarchy with phrases & entities

Input collection

i) Hierarchical topic discovery with entities

ii) Phrase mining

iii) Rank phrases & entities per topic
## 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

Input collection

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.

\[
p(z = DB | \text{query processing}) = \frac{p(z=DB)p(query | z = DB)p(\text{processing} | z = DB)}{\sum_t p(z=t)p(query | z = t)p(\text{processing} | z = t)}
\]

\[
= \frac{\theta_{DB} \Phi_{DB,query} \Phi_{DB,\text{processing}}}{\sum_t \theta_t \Phi_t,query \Phi_t,\text{processing}}
\]

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 $A$: $\alpha(A) = \frac{|A| - E(|A|)}{std(|A|)}$

$T$: topic for comparison

Significance score used for phrase mining

Pointwise KL-divergence

$$r_t(A) = \left( p(A|t) \log \frac{p(A|t)}{p(A|T)} \right) + \omega p(A|t) \log \alpha(A)$$
Example Topics: Database & Information Retrieval
Which child topic does not belong to the given parent topic?

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

Evaluation Method - Intrusion Detection
Extension of [Chang et al. 09]
% of the hierarchy interpreted by people

CS Topic Intrusion

1. hPAM
2. NetClus
3. CATHY (unigram)
3 + phrase
3 + entity

NEWS Topic Intrusion

1. hPAM
2. NetClus
3. CATHY (unigram)
3 + phrase
3 + entity
3 + phrase + entity

Phrases + Entities > Unigrams
Important research areas in SIGIR conference:

- Support vector machines
- Collaborative filtering
- Text categorization
- Text classification
- Conditional random fields
- Information systems
- Artificial intelligence
- Distributed information retrieval
- Query evaluation
- Event detection
- Large collections
- Similarity search
- Duplicate detection
- Large scale
- Information retrieval
- Question answering
- Web search
- Natural language document retrieval

Application: Entity & Community Profiling

Important research areas in SIGIR conference?
Outline

1. Introduction to bringing structure to text
2. Mining phrase-based and entity-enriched topical hierarchies
3. Heterogeneous information network construction and mining
4. Trends and research problems
Heterogeneous network construction

<table>
<thead>
<tr>
<th>Entity typing</th>
<th>Entity role analysis</th>
<th>Entity relation mining</th>
</tr>
</thead>
<tbody>
<tr>
<td>Michael Jordan – <em>researchers</em> or <em>basketball player</em>?</td>
<td>What is the role of Dan Roth/SIGIR in machine learning?Who are important <em>contributors</em> of data mining?</td>
<td>What is the relation between David Blei and Michael Jordan?</td>
</tr>
</tbody>
</table>
Type Entities from Text

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

<table>
<thead>
<tr>
<th>Type</th>
<th>Entity</th>
<th>Mention</th>
</tr>
</thead>
<tbody>
<tr>
<td>politician</td>
<td>[Obama Image]</td>
<td><em>Obama says more than 6M signed up for health care...</em></td>
</tr>
<tr>
<td>high-tech company</td>
<td>[Apple Image]</td>
<td><em>Apple leads in list of Silicon Valley's most-valuable brands...</em></td>
</tr>
</tbody>
</table>
## Large Scale Taxonomies

<table>
<thead>
<tr>
<th>Name</th>
<th>Source</th>
<th># types</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

- **Type:** 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
Type Entities in Text

- 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
  - 82 of 900 shoe brands exist in Wiki
- Sparse concept descriptors
  - Michael Jordan won the best paper award

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’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...
### Targeted Disambiguation

<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>

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

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

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

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

5. 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 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.

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

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...
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.

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 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**

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

Christos Faloutsos in data mining

Entities in Topic Hierarchy

Entity role analysis
Example Hidden Relations

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

- **Social relationship** from online social network:
  - Alumni
  - Colleague
  - Club friend

Entity relation mining

Jeffrey Naughton (1987)

Joseph M. Hellerstein (1995)

Surajit Chaudhuri (1991)

Jeff Ullman
Mining Paradigms

- Similarity search of relationships
- Classify or cluster entity relationships
- Slot filling
Similarity Search of Relationships

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

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

(Apple, iPad) $\rightarrow$ (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)

\[\text{Is advisor of}\]

\[\text{Is colleague of}\]
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>
Text Patterns

- Syntactic patterns
  - [Bunescu & Mooney 05b]

- Dependency parse tree patterns
  - [Zelenko et al. 03]
  - [Culotta & Sorensen 04]
  - [Bunescu & Mooney 05a]

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

The headquarters of Google are situated in Mountain View.

Jane says John heads XYZ Inc.

<table>
<thead>
<tr>
<th>Topic 5</th>
<th>Topic 31</th>
<th>Topic 38</th>
<th>Topic 41</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Grant Proposals”</td>
<td>“Meeting Setup”</td>
<td>“ML Models”</td>
<td>“Friendly Discourse”</td>
</tr>
<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>ill</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>

Emails between McCallum & Padhraic Smyth
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]
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
Mining Information Networks

Example: DBLP: A Computer Science bibliographic database

Knowledge hidden in DBLP Network

<table>
<thead>
<tr>
<th>Question</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**

**Similar reputation at similar venues**

<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>
Similarity Search: PathSim Measure Helps Find Peer Objects in Long Tails

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

PathSim [Sun et al. 11]

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

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

Jun Yang
- CS, Duke
- Database area
- PhD: 2001

<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>

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

The prediction power of each meta-path
Derived by logistic regression

<table>
<thead>
<tr>
<th>Meta-Path</th>
<th>Semantic Meaning</th>
<th>$p$-value</th>
<th>significance level$^1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A \rightarrow P \rightarrow P \rightarrow A$</td>
<td>$a_i$ cites $a_j$</td>
<td>0.0378</td>
<td>***</td>
</tr>
<tr>
<td>$A \rightarrow P \leftarrow P \rightarrow A$</td>
<td>$a_i$ is cited by $a_j$</td>
<td>0.0077</td>
<td>***</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>
<td>1.2974e-174</td>
<td>****</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>
<td>1.1484e-126</td>
<td>****</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>
<td>3.4867e-51</td>
<td>****</td>
</tr>
<tr>
<td>$A \rightarrow P \rightarrow P \rightarrow P \rightarrow A$</td>
<td>$a_i$ cites papers that cite $a_j$</td>
<td>0.7459</td>
<td></td>
</tr>
<tr>
<td>$A \rightarrow P \leftarrow P \leftarrow P \rightarrow A$</td>
<td>$a_i$ is cited by papers that are cited by $a_j$</td>
<td>0.0647</td>
<td>*</td>
</tr>
<tr>
<td>$A \rightarrow P \leftarrow P \rightarrow P \leftarrow A$</td>
<td>$a_i$ and $a_j$ cite the same papers</td>
<td>9.7641e-11</td>
<td>****</td>
</tr>
<tr>
<td>$A \rightarrow P \leftarrow P \rightarrow P \rightarrow A$</td>
<td>$a_i$ and $a_j$ are cited by the same papers</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:
  - IM100K: 943 items, 1360 users, 89,626 ratings, 60,905 entities, 146,013 links
  - Yelp: 11,537 items, 43,873 users, 229,907 ratings, 285,317 entities, 570,634 links

- 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>Method</th>
<th>IM100K</th>
<th>Yelp</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Prec1</td>
<td>Prec5</td>
</tr>
<tr>
<td>Popularity</td>
<td>0.0731</td>
<td>0.0513</td>
</tr>
<tr>
<td>Co-Click</td>
<td>0.0668</td>
<td>0.0558</td>
</tr>
<tr>
<td>NMF</td>
<td>0.2064</td>
<td>0.1661</td>
</tr>
<tr>
<td>Hybrid-SVM</td>
<td>0.2087</td>
<td>0.1441</td>
</tr>
<tr>
<td>HeteRec-g</td>
<td>0.2094</td>
<td>0.1791</td>
</tr>
<tr>
<td>HeteRec-p</td>
<td>0.2121</td>
<td>0.1932</td>
</tr>
</tbody>
</table>

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

1. Introduction to bringing structure to text
2. Mining phrase-based and entity-enriched topical hierarchies
3. Heterogeneous information network construction and mining
4. Trends and research problems
Mining Latent Structures from Multiple Sources

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

Topical phrase mining
Entity typing

Freebase
Satori
DBpedia
yago

Comparisons of relational database management systems - Wikipedia, the free encyclopedia

Operating system support - Comparison of relational database management systems - Grenada Wiki 6

<table>
<thead>
<tr>
<th>4th Dimension</th>
<th>Windows</th>
<th>OS X</th>
<th>Linux</th>
<th>BSD</th>
<th>UNIX</th>
<th>AIX</th>
<th>OpenVMS</th>
<th>Symbian</th>
<th>z/OS</th>
<th>iOS</th>
<th>Android</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACIDAS</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Adaptive Server Enterprise</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Advantage Database Server</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Alfamante</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>

Isolation (database systems) - Wikipedia, the free encyclopedia

<table>
<thead>
<tr>
<th>Isolation level</th>
<th>Write Operation</th>
<th>Read Operation</th>
<th>Range Operation (...where...)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read Uncommitted</td>
<td>$S$</td>
<td>$S$</td>
<td>$S$</td>
</tr>
<tr>
<td>Read Committed</td>
<td>$C$</td>
<td>$S$</td>
<td>$S$</td>
</tr>
<tr>
<td>Repeatable Read</td>
<td>$C$</td>
<td>$C$</td>
<td>$S$</td>
</tr>
<tr>
<td>Satisfiable</td>
<td>$C$</td>
<td>$C$</td>
<td>$C$</td>
</tr>
</tbody>
</table>
Integration of NLP & Data Mining

NLP - analyzing single sentences

Data mining - analyzing big data

Topical phrase mining
Entity typing

Example diagram with graph and labeled nodes.
Open Problems on Mining Latent Structures

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

- 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?
Understand the People

- NLP, ML, AI
  Understand & answer natural language questions

- HCI, Crowdsourcing, Web search, domain experts
  Explore latent structures with user guidance
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


References


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


References


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


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


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.


References


