Representing Documents via Latent Keyphrase Inference

Jialu Liu†  Xiang Ren†  Jingbo Shang†
Taylor Cassidy†  Clare R. Voss†  Jiawei Han†
†Department of Computer Science, University of Illinois at Urbana-Champaign
‡Computational & Information Sciences Directorate, Army Research Laboratory

April. 15th, 2016
Document Representation in Vector Space

Critical for document retrieval, categorization
Traditional Methods

- Bag-of-Words or Phrases

<table>
<thead>
<tr>
<th></th>
<th>doc1</th>
<th>doc2</th>
<th>doc3</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>like</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>football</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>John</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>likes</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>football</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>basketball</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

- Cons: Sparse on short texts
Topic models [LDA]

Each topic is a distribution over words, each document is a mixture of corpus-wide topics

Cons: Difficult for human to infer topic semantics
- Concept-based models [ESA]

Every Wikipedia article represents a **concept**

Panthera

From Wikipedia, the free encyclopedia

Panthera is a genus of the family Felidae (the cats), which contains four well-known living species: the lion, tiger, jaguar, and leopard. The genus comprises about half of the big cats. One meaning of the word *panther* is to designate cats of this family. Only these four cat species have the anatomical changes enabling them to roar. The primary reason for this was assumed to be the incomplete ossification of the hyoid bone. However, new studies show that the ability to roar is due to other morphological features, especially of the larynx. The snow leopard *Uncia uncia*, which is sometimes included within Panthera, does not roar. Although it has an incomplete ossification of the hyoid bone, it lacks the special morphology of the larynx, which is typical for lions, tigers, jaguars and leopards.

Species and subspecies

- [edit]

Article words are associated with the concept (TF.IDF), which help infer concepts from document

- Cons: Low coverage of concepts in human-curated knowledge base
- Word/Document embedding models [word2vec paragraph2vec]

- Cons: Difficult to explain what each dimension means
Use **domain keyphrases** as the entries in the vector and

Identify **document keyphrases** (subset of domain keyphrases) by evaluating relatedness between (doc, domain keyphrase)

Unsupervised model
Challenges

- Where to get domain keyphrases from a given corpus?
  - Mining Quality Phrases from Massive Text Corpora [SIGMOD15]

- How to identify document keyphrases?
  - Can be latent mentions (short text)
  - Relatedness scores
How to identify document keyphrases?

- Powered by Bayesian Inference on “Domain Keyphrase Silhouette”
  - Domain Keyphrase Silhouette: Topic centered on domain keyphrase
    - “Reverse” topic models
    - Learned from corpus

---

**Kernel k-means**
- kernel kmeans 1
- kernel kmeans 1
- clustering 0.65
- kernel 0.55
- rbf kernel 0.5

**DBSCAN**
- dbscan 1
- density 0.8
- clustering 0.6
- dense regions 0.3
- shape 0.25

**Data Mining**
- data mining 1
- knowl. discov. 1
- kdd 0.67
- clustering 0.6
- text mining 0.6
Framework for Latent Keyphrase Inference (LAKI)

Keyphrase Extraction

**Offline:**
- `data mining`
- `text mining`
- `clustering`
- `kernel k-means`
- `dbscan`

DBSCAN / is / a / method / for / clustering / in / process / of / knowledge discovery. DBSCAN / was / proposed / by ...

Segmentation

**Online:**

```
  data mining
  kernel k-means
  dbscan
```

```
  kernel k-means
  clustering 0.65
  kernel 0.55
  rbf kernel 0.5
```

```
  dbscan
  density 0.8
  clustering 0.6
  dense regions 0.3
  shape 0.25
```

```
  data mining
  knowl. discov. 1
  kdd 0.67
  clustering 0.6
  text mining 0.6
```

Keyphrase Silhouetting

```
  data mining
  clustering
  dbscan
```

```
  data mining
  knowledge discovery
  kdd
  dbscan
  clustering
  data
  kernel k-means
```

```
  data mining
  clustering
  dbscan
```

```
  kernel k-means
  knowledge discovery
  density-based clustering
```

Document Representation
Keyphrase Extraction
- data mining
- text mining
- clustering
- kernel k-means
- dbscan

Offline:
DBSCAN / is / a / method / for / clustering / in / process / of / knowledge discovery. DBSCAN / was / proposed / by ...

Segmentation

Keyphrase Silhouetting
- kernel k-means
- dbscan
- data mining

Inference

Document Representation
- data mining
- clustering
- dbscan

Segmentation
- knowledge discovery
- kdd
- dbscan
- clustering
- data
- kernel k-means
Domain Keyphrase Silhouette

- Learning Hierarchical Bayesian Network (DAG)

Task 1: Model Learning: learning link weights
Task 2: Structure Learning: learning network structure
Task 1: Model Learning given Structure

- Use $Z$ to represent $K$ (domain keyphrases) and $T$ (content units)
- Noisy-OR
  - A parent node is easier to activate its children when the link weight is larger
  - A child node is influenced by all its parents

\[
p(Z_j = 1 \mid Pa(Z_j)) = 1 - \exp\left(-W_{0j} - \sum_i W_{ij} 1_{Pa_i^j}\right)
\]

- Noise / Prior
- Aggregated over all other links connected with $Z_j$
Maximum Likelihood Estimation

- Training data: Documents
- Expectation-step:
  - For each document, collect sufficient statistics
  - Link firing (Parent, child both being activated) probability
  - Node activation probability
- Maximization-step:
  - Update link weight

\[
L(D) = \sum_{d=1}^{N} \log \sum_{k \in \Omega^{(d)}} p(K = k, T = t^{(d)})
\]
Task 2: Structure Learning

- Domain keyphrases are connected to content units
  - Help infer document keyphrases from content units

- Domain keyphrases are interconnected
  - Help infer document keyphrases from other keyphrases
A Heuristic Approach

- Data-Driven, DAG, similar to ontology
- Heuristic:
  - Two nodes are connected only
    - Closely Related: word2vec
    - Co-occur frequently
  - Links are always point to less frequent nodes
- Work well in practice
Keyphrase Extraction

Offline:
- data mining
- text mining
- clustering
- kernel k-means
- dbscan

DBSCAN / is / a / method / for / clustering / in / process / of / knowledge discovery.
DBSCAN / was / proposed / by ...

Segmentation

Keyphrase Silhouetting

Inference

Document Representation
Inference

- Exact inference is slow!
  - NP hard to compute posterior probability for Noisy-Or networks
- Approximate inference instead
  - Pruning irrelevant nodes using an efficient scoring function
  - Gibbs sampling
Experiments

- Two text-related tasks to evaluate document representation quality
  - Phrase relatedness
  - Document classification

- Two datasets

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#Docs</th>
<th>#Words</th>
<th>Content type</th>
<th>#Domain Keyphr. (in Wiki)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Academia</td>
<td>0.43M</td>
<td>28M</td>
<td>title &amp; abstract</td>
<td>33K (6,367)</td>
</tr>
<tr>
<td>Yelp</td>
<td>0.47M</td>
<td>98M</td>
<td>review</td>
<td>25K (4,996)</td>
</tr>
</tbody>
</table>
Methods

- **ESA** (Explicit Semantic Analysis)
- **KBLLink** uses link structure in Wikipedia
- **BoW** (bag-of-words)
- **ESA-C** extends ESA by replacing Wiki with domain corpus
- **LSA** (Latent Semantic Analysis)
- **LDA** (Latent Dirichlet Allocation)
- **Word2Vec** is a neural network computing word embeddings
- **EKM** uses explicit keyphrase detection

<table>
<thead>
<tr>
<th>Method</th>
<th>Semantic Space</th>
<th>Input Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESA</td>
<td>KB concepts</td>
<td>KB</td>
</tr>
<tr>
<td>KBLLink</td>
<td>KB concepts</td>
<td>KB</td>
</tr>
<tr>
<td>BoW</td>
<td>Words</td>
<td>-</td>
</tr>
<tr>
<td>ESA-C</td>
<td>Documents</td>
<td>Corpus</td>
</tr>
<tr>
<td>LSA</td>
<td>Topics</td>
<td>Corpus</td>
</tr>
<tr>
<td>LDA</td>
<td>Topics</td>
<td>Corpus</td>
</tr>
<tr>
<td>Word2Vec</td>
<td>-</td>
<td>Corpus</td>
</tr>
<tr>
<td>EKM</td>
<td>Explicit Keyphrases</td>
<td>Corpus</td>
</tr>
<tr>
<td>LAKI</td>
<td>Latent Keyphrases</td>
<td>Corpus</td>
</tr>
</tbody>
</table>
### Phrase Relatedness Correlation

<table>
<thead>
<tr>
<th>Method</th>
<th>Academia (w/ phrase)</th>
<th>Yelp (w/ phrase)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESA</td>
<td>0.4320 (-)</td>
<td>0.4567 (-)</td>
</tr>
<tr>
<td>KBLink</td>
<td>0.1878 (-)</td>
<td>0.4179 (-)</td>
</tr>
<tr>
<td>ESA-C</td>
<td>0.4905 (0.5243)</td>
<td>0.4655 (0.5029)</td>
</tr>
<tr>
<td>LSA</td>
<td>0.5877 (0.6383)</td>
<td>0.6700 (0.7229)</td>
</tr>
<tr>
<td>LDA</td>
<td>0.3610 (0.5391)</td>
<td>0.3928 (0.5405)</td>
</tr>
<tr>
<td>Word2Vec</td>
<td>0.6674 (0.7281)</td>
<td>0.7143 (0.7419)</td>
</tr>
<tr>
<td>LAKI</td>
<td><strong>0.7504</strong></td>
<td><strong>0.7609</strong></td>
</tr>
</tbody>
</table>

### Document Classification

<table>
<thead>
<tr>
<th>Method</th>
<th>Academia (w/ phrase)</th>
<th>Yelp (w/ phrase)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESA</td>
<td>37.61 (-)</td>
<td>46.56 (-)</td>
</tr>
<tr>
<td>KBLink</td>
<td>36.37 (-)</td>
<td>35.94 (-)</td>
</tr>
<tr>
<td>BoW</td>
<td>48.05 (45.60)</td>
<td>51.26 (45.97)</td>
</tr>
<tr>
<td>ESA-C</td>
<td>39.75 (42.20)</td>
<td>49.13 (54.51)</td>
</tr>
<tr>
<td>LSA</td>
<td>72.50 (79.22)</td>
<td>66.55 (78.57)</td>
</tr>
<tr>
<td>LDA</td>
<td>77.27 (80.52)</td>
<td>75.55 (82.65)</td>
</tr>
<tr>
<td>EKM</td>
<td>45.46</td>
<td>40.57</td>
</tr>
<tr>
<td>LAKI</td>
<td><strong>84.42</strong></td>
<td><strong>90.58</strong></td>
</tr>
</tbody>
</table>
# Case Study

<table>
<thead>
<tr>
<th>Query</th>
<th>LDA</th>
<th>BOA</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Keyphrases</strong></td>
<td>linear discriminant analysis, latent dirichlet allocation, topic models, topic modeling, face recognition, sda, latent dirichlet, generative model, topic, subspace models, ...</td>
<td>boa steakhouse, bank of america, stripsteak, agnolotti, credit card, santa monica, restaurants, wells fargo, steakhouse, prime rib, bank, vegas, las vegas, cash, cut, dinner, bank, money, ...</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Query</th>
<th>LDA topic</th>
<th>BOA steak</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Keyphrases</strong></td>
<td>latent dirichlet allocation, topic, topic models, topic modeling, probabilistic topic models, latent topics, topic discovery, generative model, mixture, text mining, topic distribution, plsi, ...</td>
<td>steak, stripsteak, boa steakhouse, steakhouse, ribeye, craftsteak, santa monica, medium rare, prime, vegas, entrees, potatoes, french fries, filet mignon, mashed potatoes, texas roadhouse, ...</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Query</th>
<th>SVM</th>
<th>deep dish pizza</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Keyphrases</strong></td>
<td>support vector machines, svm classifier, multi class, training set, margin, knn, classification problems, kernel function, multi class svm, multi class support vector machine, support vector, ...</td>
<td>deep dish pizza, chicago, deep dish, amore taste of chicago, amore, pizza, oregano, chicago style, chicago style deep dish pizza, thin crust, windy city, slice, pan, oven, pepperoni, hot dog, ...</td>
</tr>
<tr>
<td>Query</td>
<td>Mining Frequent Patterns without Candidate Generation</td>
<td>I am a huge fan of the All You Can Eat Chinese food buffet.</td>
</tr>
<tr>
<td>-------------------------------------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Keyphrases</td>
<td>mining frequent patterns, candidate generation, frequent pattern mining, candidate, prune, fp growth, frequent pattern tree, apriori, subtrees, frequent patterns, candidate sets, ...</td>
<td>all you can eat, chinese food, buffet, chinese buffet, dim sum, orange chicken, chinese restaurant, asian food, asian buffet, crab legs, lunch buffet, fan, salad bar, all you can drink, ...</td>
</tr>
</tbody>
</table>

| Query                                           | Text mining, also referred to as text data mining, roughly equivalent to text analytics, refers to the process of deriving high-quality information from text. High-quality information is typically derived through means such as statistical pattern learning. | It's the perfect steakhouse for both meat and fish lovers. My table guest was completely delirious about his Kobe Beef and my lobster was perfectly cooked. Good wine list, they have a lovely Sancerre! Professional staff, quick and smooth. |
| Keyphrases                                      | text analytics, text mining, patterns, text, textual data, topic, information, text documents, information extraction, machine learning, data mining, knowledge discovery, ... | kobe beef, fish lovers, steakhouse, sancerre, wine list, guests, perfectly cooked, lobster, staff, meat, fillet, fish, lover, seafood, ribeye, filet, sea bass, risotto, starter, scallops, steak, beef, ... |
Time Complexity

- Plot 1: Running Time vs. #Samples
- Plot 2: Running Time vs. #Quality Phrases After Pruning
- Plot 3: Running Time vs. #Words

Graphs show trends in running time for different datasets (Academia, Yelp) across varying sample sizes and other metrics.
Breakdown of Processing Time

- **50-word Query**
  - Sampling: 70.6%
  - Pruning: 29.2%
  - Segmentation: 0.2%

- **400-word Query**
  - Sampling: 87.1%
  - Pruning: 12.7%
  - Segmentation: 0.2%
Conclusion

- We have introduced a novel document representation method using **latent keyphrases**
  - Each dimension is explainable
  - Works for short text
  - Works for closed-domain text
- We have developed an efficient inference method to do **real time keyphrase identification**
- Future work
  - Better structure learning approach
  - Combined with knowledge base
  - Try other inference method other than Gibbs sampling
- Code available at http://jialu.info