Maps of Computer Science

Daniel Fried and Stephen G. Kobourov

Department of Computer Science,
University of Arizona,
http://mocs.cs.arizona.edu
Creating Maps from Paper Titles

- Graph vertices ("cities"): terms representing research topics
- Graph edges ("roads"): term similarity, co-occurrence
- Vertex clusters ("countries"): generally reflect research areas
Dataset: The DBLP bibliography server (DataBase systems and Logic Programming)
Dataset: The DBLP bibliography server (DataBase systems and Logic Programming)
- covers most CS journals/conf. (about 6,000 different ones)
- over 2.1 million indexed publications
- includes titles and bibliographic information
Dataset: The DBLP bibliography server (DataBase systems and Logic Programming)
- covers most CS journals/conf. (about 6,000 different ones)
- over 2.1 million indexed publications
- includes titles and bibliographic information

Main problems:
- large dataset (448,374 different words; 2,089,736 phrases)
- short text (only titles, with 10 words on average)
The MoCS System

Term Extraction

Choose docs for basemap
Choose docs for heatmap

POS Tag Matching

Multi-word or Single-word Terms

baseset terms

algorithms

seam carving

support vector machines

databases

computational complexity

heatmap terms

algorithms

support vector machines

databases

Term Ranking and Similarity

Ranking

Similarity

TF/ICF or C-value

Jaccard or LSA

Pairwise Similarities

Filtered Similarities

top terms

or pull lesser

Frequency Count

Term Intersection

Heatmap Overlay

Basemap

Term Layout

Map Creation

Euclidean Distances

logarithmic scaling

Filtered Similarities

top terms

or pull lesser

MDS
Term Extraction

Choose docs for basemap

Choose docs for heatmap

POS Tag Matching

<NN>|<JJ>*

Multi-word or Single-word Terms

basemap terms
algorithms
seam carving
support vector machines
databases
computational complexity

heatmap terms
algorithms
support vector machines
databases
computational complexity

Term Ranking and Similarity

Pairwise Similarities

basemap terms
heatmap terms

TF/ICF or C-value
Jaccard or LSA

Filtering and Distance Scaling

Filtered Similarities

Euclidean Distances

Ranking

Term Layout

Basemap

Term Intersection

Heatmap Overlay

Frequency Count

MDS

Clustering

Top terms or pull lesser
Term Extraction

Multi-word phrases ("collocations")

- Specificity: "wireless sensor networks" as a type of "network"
- Context: "travelling salesman problem", not "salesman"
- POS tagging and filtering - Justeson and Katz, 1995
  
<table>
<thead>
<tr>
<th>POS</th>
<th>NNS</th>
<th>IN</th>
<th>JJ</th>
<th>NN</th>
<th>NN</th>
</tr>
</thead>
<tbody>
<tr>
<td>word</td>
<td>applications of wireless sensor networks</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Extract noun and adjective subsequences
- Multi-word, or break up into single words
Term Ranking and Similarity

Top terms
- algorithms
- seam carving
- databases
- computational complexity

Top Terms
- algorithms
- seam carving
- databases
- computational complexity

Ranking
- TF/ICF or C-value
- Jaccard or LSA

Similarity
- Pairwise Similarities
- Euclidean Distances
- Logarithmic scaling

Map Creation
- Term Ranking
- Similarity
- MDS Clustering
- Term Layout
- Heatmap Overlay
- Frequency Count
- Top terms
- or pull lesser
Term Ranking

- Simplest possible ranking: by frequency
Simplest possible ranking: by frequency

TF-IDF: term frequency – inverse document frequency
  – Extra difficult due to short titles (IDF is meaningless)
Term Ranking

- Simplest possible ranking: by frequency
- TF-IDF: term frequency – inverse document frequency
  - Extra difficult due to short titles (IDF is meaningless)
- TF-ICF: term frequency – inverse corpus frequency
  - Expensive
Term Ranking

- Simplest possible ranking: by frequency
- TF-IDF: *term frequency* – *inverse document frequency*
  – Extra difficult due to short titles (IDF is meaningless)
- TF-ICF: *term frequency* – *inverse corpus frequency*
  – Expensive
- Best results: C-Value - Frantzi et al, 2000
  1. Term frequency: +
  2. Length of the term: +
  3. Occurrences nested in other terms: -
  4. Number of these other terms: +
Term Similarity: LSA and Cosine

- Term-document matrix $A$
- Latent Semantic Analysis (LSA) - decompose $A$
- Cosine distance - compare angles

$$\text{Dist}(T_i, T_j) = \frac{T_i \cdot T_j}{||T_i|| \ ||T_j||}$$

- Small angle (large cosine): similar terms
- Large angle (small cosine): dissimilar terms
Term Similarity: Jaccard Coefficient

- Idea: terms are similar if they are used together in titles
- Treat as set similarity: $S_i$ is the set of documents with term $i$
- Jaccard coefficient:

$$Jacc(S_i, S_j) = \frac{|S_i \cap S_j|}{|S_i \cup S_j|}$$

- Extra difficult due to multi-word terms
- Partial match Jaccard: count co-occurrence if terms overlap
Filtering and Distance Scaling

- LSA and Jaccard return similarity values between 0 and 1
- Convert to distances for graph drawing
- Inverse logarithmic spacing
- Top Terms: only plot $N$ highest-ranked terms
- Pull Lesser Terms: plot $K$ most similar terms for each term $t$
Making a Map with GMap

- Input: vertex-weighted, edge-weighted graph $G = (V, E)$
- Output: map, with clusters as countries and vertices as cities
- GMap: a framework for embedding + clustering + mapping
  - different algorithms: embedding, clustering, mapping
  - different overlays: journal profile, author profile, paper profile
GMap Overview

- **Embedding**
  - *scalable force-directed method*
  - iterative improvement
  - minimal energy $\Rightarrow$ good layout

- **Clustering**
  - *modularity clustering*
  - group vertices such that:
    - high edge density *within* groups
    - low edge density *between* groups

- **Mapping**
  - *modified Voronoi Diagram*
  - add bounding box
  - add dummy points to get nice borders
Base Map of CS
Base Map of CS
Base Map of CS
Base Map of CS
Base Map of CS
Heatmap Profiles

- Visualize an author, conference, journal, or timeframe
- Want to see intensity of term usage and spread over the map
- Extract terms in same way as basemap
- Count frequencies of term intersection

\[ \hat{l}(t) = \frac{\log(tf(t) + \beta)}{\max_{\hat{t}} \log(tf(\hat{t}) + \beta)} \]

\( tf(t) \): frequency of term \( t \) in heatmap query

\( \beta \): small constant
Computer Vision and Pattern Recognition
Using DBLP Metadata

- Separate queries for basemaps and heatmaps
- DBLP metadata allows query variation
  - by venue: 1,324 journals; 6,904 conferences
  - by author: 1,237,445 authors
  - by date: 1950 - present
- Visualize authors in the context of their venues
- Visualize change in a venue’s research focus over time
Temporal Heatmaps: JACM 1954-1963
Temporal Heatmaps: JACM 1954-1963
Temporal Heatmaps: JACM 1974-1983
Temporal Heatmaps: JACM 1984-1993
Temporal Heatmaps: JACM 2004-2013
Can vary basemap and heatmap queries independently

Runtime varies: a few seconds for an author, about a minute for 60,000 doc sample of all papers

Open source, modular, extensible – add your own term similarity, ranking, etc. functions:
github.com/dpfried/mocs

Interactive web interface: mocs.cs.arizona.edu
Future Work

- Dealing with sparsity: using abstracts and full papers
- Reducing map fragmentation with contiguous country maps
- Try on paper corpora from other domains
  - PubMed
  - arXiv
- Map validation: consistency and recall (expert evaluation)
Thanks!