Content Selection: Graphs, Supervision, HMMs

Ling573
Systems & Applications
April 6, 2017
Roadmap

- MEAD: classic end-to-end system
  - Cues to content extraction
- Bayesian topic models
- Graph-based approaches
  - Random walks
- Supervised selection
  - Term ranking with rich features
MEAD

- Exemplar centroid-based summarization system
  - Tf-idf similarity measures
- Multi-document summarizer
- Publically available summarization implementation
  - (No warranty)
- Solid performance in DUC evaluations
- Standard non-trivial evaluation baseline
Main Ideas

- Select sentences central to cluster:
  - Cluster-based relative utility
    - Measure of sentence relevance to cluster

- Select distinct representative from equivalence classes
  - Cross-sentence information subsumption
    - Sentences including same info content said to subsume
      - A) John fed Spot; B) John gave food to Spot and water to the plants.
        - I(B) subsumes I(A)
        - If mutually subsume, form equivalence class
Centroid-based Models

- Assume clusters of topically related documents
  - Provided by automatic or manual clustering

- Centroid: “pseudo-document of terms with Count * IDF above some threshold”
  - Intuition: centroid terms indicative of topic
  - Count: average # of term occurrences in cluster
  - IDF computed over larger side corpus (e.g. full AQUAINT)
MEAD Content Selection

- **Input:**
  - Sentence segmented, cluster documents (n sents)
  - Compression rate: e.g. 20%

- **Output:** n * r sentence summary

- Select highest scoring sentences based on:
  - Centroid score
  - Position score
  - First-sentence overlap
  - (Redundancy)
Score Computation

- Score\( (s_i) = w_c C_i + w_p P_i + w_f F_i \)
  - \( C_i = \sum \Sigma_{i} C_{w,i} \)
    - Sum over centroid values of words in sentence
  - \( P_i = ((n-i+1)/n) * C_{\text{max}} \)
    - Positional score: \( C_{\text{max}} \): score of highest sent in doc
      - Scaled by distance from beginning of doc
  - \( F_i = S_1 * S_i \)
    - Overlap with first sentence
    - TF-based inner product of sentence with first in doc

- Alternate weighting schemes assessed
  - Diff’t optima in different papers
Managing Redundancy

- Alternative redundancy approaches:
  - Redundancy max:
    - Excludes sentences with cosine overlap > threshold
  - Redundancy penalty:
    - Subtracts penalty from computed score
    - \( R_s = 2 \times \# \text{overlapping wds}/(\# \text{wds in sentence pair}) \)
      - Weighted by highest scoring sentence in set
System and Evaluation

• Information ordering:
  • Chronological by document date

• Information realization:
  • Pure extraction, no sentence revision

• Participated in DUC 2001, 2003
  • Among top-5 scoring systems
  • Varies depending on task, evaluation measure

• Solid straightforward system
  • Publicly available; will compute/output weights
Bayesian Topic Models

- Perspective: Generative story for document topics
- Multiple models of word probability, topics
  - General English
  - Input Document Set
  - Individual documents
- Select summary which minimizes KL divergence
  - Between document set and summary: $\text{KL}(P_D || P_S)$
- Often by greedily selecting sentences
  - Also global models
Graph-Based Models

- LexRank (Erkan & Radev, 2004)

- Key ideas:
  - Graph-based model of sentence saliency
    - Draws ideas from PageRank, HITS, Hubs & Authorities
  - Contrasts with straight term-weighting models
  - Good performance: beats tf*idf centroid
Graph View

- Centroid approach:
  - Central pseudo-document of key words in cluster

- Graph-based approach:
  - Sentences (or other units) in cluster link to each other
  - Salient if similar to many others
    - More central or relevant to the cluster
  - Low similarity with most others, not central
Constructing a Graph

- Graph:
  - Nodes: sentences
  - Edges: measure of similarity between sentences

- How do we compute similarity b/t nodes?
  - Here: tf*idf (could use other schemes)

- How do we compute overall sentence saliency?
  - Degree centrality
  - LexRank
Example Graph
Degree Centrality

- Centrality: # of neighbors in graph
  - Edge\((a,b)\) if \(\text{cosine\_sim}(a,b) \geq \text{threshold}\)

- Threshold = 0:
  - Fully connected \(\rightarrow\) uninformative

- Threshold = 0.1, 0.2:
  - Some filtering, can be useful

- Threshold \(\geq 0.3\):
  - Only two connected pairs in example
  - Also uninformative
LexRank

- Degree centrality: 1 edge, 1 vote
  - Possibly problematic:
    - E.g. erroneous doc in cluster, some sent. may score high

- LexRank idea:
  - Node can have high(er) score via high scoring neighbors
    - Same idea as PageRank, Hubs & Authorities
      - Page ranked high b/c pointed to by high ranking pages

\[
p(u) = \sum_{v \in \text{adj}(u)} \frac{p(v)}{\text{deg}(v)}
\]
Power Method

- **Input:**
  - Adjacency matrix $M$

- Initialize $p_0$ (uniform)

- $t=0$

- repeat
  - $t= t+1$
  - $p_t= M^T p_{t-1}$

- Until convergence

- Return $p_t$
LexRank

- Can think of matrix $X$ as transition matrix of Markov chain
  - i.e. $X(i,j)$ is probability of transition from state $i$ to $j$

- Will converge to a stationary distribution ($r$)
  - Given certain properties (aperiodic, irreducible)
  - Probability of ending up in each state via random walk

- Can compute iteratively to convergence via:

$$p(u) = \frac{d}{N} + (1 - d) \sum_{v \in adj(u)} \frac{p(v)}{\text{deg}(v)}$$

  - “Lexical PageRank” $\Rightarrow$ “LexRank
  - (power method computes eigenvector)
LexRank Score Example

- For earlier graph:

<table>
<thead>
<tr>
<th>ID</th>
<th>LR (0.1)</th>
<th>LR (0.2)</th>
<th>LR (0.3)</th>
<th>Centroid</th>
</tr>
</thead>
<tbody>
<tr>
<td>d1s1</td>
<td>0.6007</td>
<td>0.6944</td>
<td>1.0000</td>
<td>0.7209</td>
</tr>
<tr>
<td>d2s1</td>
<td>0.8466</td>
<td>0.7317</td>
<td>1.0000</td>
<td>0.7249</td>
</tr>
<tr>
<td>d2s2</td>
<td>0.3491</td>
<td>0.6773</td>
<td>1.0000</td>
<td>0.1356</td>
</tr>
<tr>
<td>d2s3</td>
<td>0.7520</td>
<td>0.6550</td>
<td>1.0000</td>
<td>0.5694</td>
</tr>
<tr>
<td>d3s1</td>
<td>0.5907</td>
<td>0.4344</td>
<td>1.0000</td>
<td>0.6331</td>
</tr>
<tr>
<td>d3s2</td>
<td>0.7993</td>
<td>0.8718</td>
<td>1.0000</td>
<td>0.7972</td>
</tr>
<tr>
<td>d3s3</td>
<td>0.3548</td>
<td>0.4993</td>
<td>1.0000</td>
<td>0.3328</td>
</tr>
<tr>
<td>d4s1</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.9414</td>
</tr>
<tr>
<td>d5s1</td>
<td>0.5921</td>
<td>0.7399</td>
<td>1.0000</td>
<td>0.9580</td>
</tr>
<tr>
<td>d5s2</td>
<td>0.6910</td>
<td>0.6967</td>
<td>1.0000</td>
<td>1.0000</td>
</tr>
<tr>
<td>d5s3</td>
<td>0.5921</td>
<td>0.4501</td>
<td>1.0000</td>
<td>0.7902</td>
</tr>
</tbody>
</table>
Continuous LexRank

- Basic LexRank ignores similarity scores
  - Except for initial thresholding of adjacency
- Could just use weights directly (rather than degree)

\[
p(u) = \frac{d}{N} + (1 - d) \sum_{v \in \text{adj}(u)} \frac{\cos \text{sim}(u, v)}{\sum_{z \in \text{adj}(v)} \cos \text{sim}(z, v)} p(v)
\]
Advantages vs Centroid

- Captures information subsumption
  - Highly ranked sentences have greatest overlap with adjacent
  - Will promote those sentences

- Reduces impact of spurious high-IDF terms
  - Rare terms get very high weight (reduce TF)
  - Lead to selection of sentences with high IDF terms
  - Effect minimized in LexRank
Example Results

- Beat official DUC 2004 entrants:
- All versions beat baselines and centroid

<table>
<thead>
<tr>
<th></th>
<th>2004 Task2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>min</td>
</tr>
<tr>
<td>Centroid</td>
<td>0.3580</td>
</tr>
<tr>
<td>Degree (t=0.1)</td>
<td>0.3590</td>
</tr>
<tr>
<td>LexRank (t=0.1)</td>
<td>0.3646</td>
</tr>
<tr>
<td>Cont. LexRank</td>
<td>0.3617</td>
</tr>
<tr>
<td>baselines:</td>
<td>random:</td>
</tr>
<tr>
<td>lead-based:</td>
<td>0.3686</td>
</tr>
</tbody>
</table>
Example Results

- Beat official DUC 2004 entrants:
  - All versions beat baselines and centroid
  - Continuous LR > LR > degree
  - Variability across systems/tasks

<table>
<thead>
<tr>
<th></th>
<th>2004 Task2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>min</td>
</tr>
<tr>
<td>Centroid</td>
<td>0.3580</td>
</tr>
<tr>
<td>Degree (t=0.1)</td>
<td>0.3590</td>
</tr>
<tr>
<td>LexRank (t=0.1)</td>
<td>0.3646</td>
</tr>
<tr>
<td>Cont. LexRank</td>
<td>0.3617</td>
</tr>
</tbody>
</table>

baselines: random: 0.3238
lead-based: 0.3686
(b)
Example Results

- Beat official DUC 2004 entrants:
  - All versions beat baselines and centroid
  - Continuous LR > LR > degree
    - Variability across systems/tasks

<table>
<thead>
<tr>
<th></th>
<th>2004 Task2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>min</td>
</tr>
<tr>
<td>Centroid</td>
<td>0.3580</td>
</tr>
<tr>
<td>Degree (t=0.1)</td>
<td>0.3590</td>
</tr>
<tr>
<td>LexRank (t=0.1)</td>
<td>0.3646</td>
</tr>
<tr>
<td>Cont. LexRank</td>
<td>0.3617</td>
</tr>
</tbody>
</table>

- Common baseline and component