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

- Radev et al, 2000, 2001, 2004
- 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

- $Score(s_i) = w_c C_i + w_p P_i + w_f F_i$
  - $C_i = \Sigma_i C_{w,I}$
    - Sum over centroid values of words in sentence
  - $P_i = ((n-i+1)/n) * C_{max}$
    - Positional score: C_{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:
  - Redundancymax:
    - Excludes sentences with cosine overlap > threshold
  - Redundancy penalty:
    - Subtracts penalty from computed score
      - R_s = 2 * # overlapping wds/(# 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: $KL(P_D || P_S)$
- 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

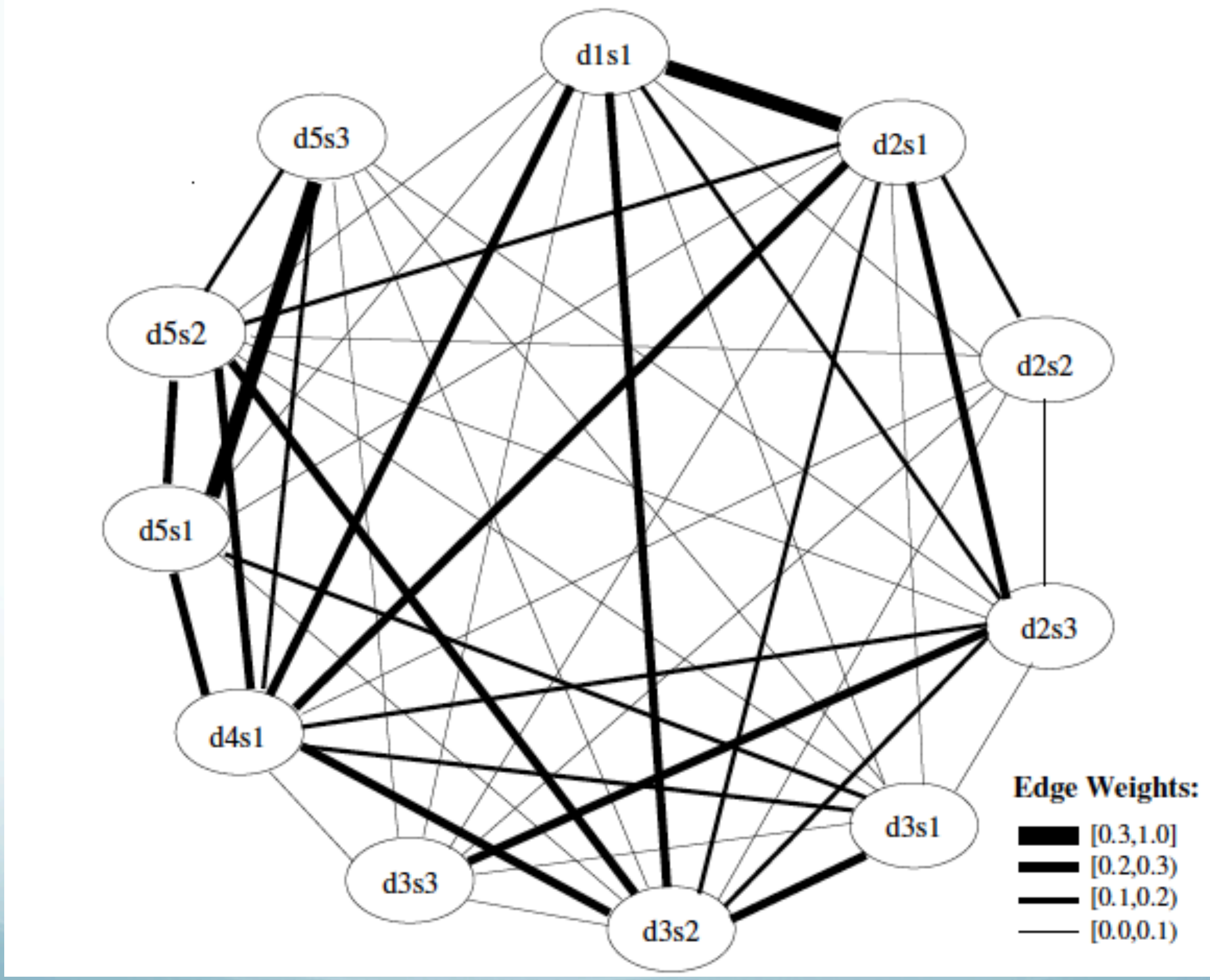
- Centroid approach:
  - Central pseudo-document of key words in cluster

- 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 cosine_sim(a,b) >= threshold
- Threshold = 0:
  - Fully connected → uninformative
- Threshold = 0.1, 0.2:
  - Some filtering, can be useful
- Threshold >= 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
    - $$p(u) = \sum_{v \in adj(u)} \frac{p(v)}{\deg(v)}$$

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:

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

  - "Lexical PageRank" ➔ "LexRank
  - (power method computes eigenvector)

LexRank Score Example

- For earlier graph:

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

Continuous LexRank

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

$$p(u) = \frac{d}{N} + (1-d) \sum_{v \in adj(u)} \frac{\cos sim(u,v)}{\sum_{z \in adj(v)} \cos sim(z,v)} p(v)$$

Advantages vs Centroid

- Captures information subsumption
  - Highly ranked sentences have greatest overlap w/adj
  - Will promote those sentences
- Reduces impact of spurious high-IDF terms
  - Rare terms get very high weight (reduce TF)
  - Lead to selection of sentences w/high IDF terms
  - Effect minimized in LexRank

Example Results

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

	2004 Task2		
	min	max	average
Centroid	0.3580	0.3767	0.3670
Degree (t=0.1)	0.3590	0.3830	0.3707
LexRank (t=0.1)	0.3646	0.3808	0.3736
Cont. LexRank	0.3617	0.3826	0.3758

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

	2004 Task2		
	min	max	average
Centroid	0.3580	0.3767	0.3670
Degree (t=0.1)	0.3590	0.3830	0.3707
LexRank (t=0.1)	0.3646	0.3808	0.3736
Cont. LexRank	0.3617	0.3826	0.3758

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

	2004 Task2		
	min	max	average
Centroid	0.3580	0.3767	0.3670
Degree (t=0.1)	0.3590	0.3830	0.3707
LexRank (t=0.1)	0.3646	0.3808	0.3736
Cont. LexRank	0.3617	0.3826	0.3758
baselines:	random:	0.3238	
	lead-based:	0.3686	

(b)

  - Common baseline and component