Topic-Orientation & Optimization

Ling573
Systems & Applications
April 18, 2017

Roadmap

- Topic-focused summarization
  - Focusing existing approaches
    - LexRank
    - CLASSY, FastSum
- Summarization with LSA
- Summarization as optimization
- Information Ordering:
  - Basic approaches
    - Variants on chronological ordering
  - Enhancing cohesion

Key Idea

- Topic-focused summarization
  - (aka ”query-focused”, “guided”)
- Motivations:
  - Extrinsic task vs generic
    - Why are we creating this summary?
      - Viewed as complex question answering (vs factoid)
  - High variation in human summaries
    - Depending on perspective different content focused
- Idea:
  - Target response to specific question, topic in docs
    - Later TACs identify topic categories and aspects
      - E.g Natural disasters: who, what, where, when..

Query-focused LexRank

- Focus on sentences relevant to query
  - Rather than uniform jump
- How do we measure relevance?
  - Tf*idf-like measure over sentences & query
    - Compute sentence-level "idf"
      - N = # of sentences in cluster; sf_w = # of sentences with w

$$idf_w = \log\left(\frac{N+1}{0.5+sf_w}\right)$$

$$rel(s|q) = \sum_{w \in q} \log(tf_{w,s}+1) * \log(tf_{w,q}+1) * idf_w$$

Updated LexRank Model

- Combines original similarity weighting w/query
  - Mixture model of query relevance, sentence similarity

$$p(s|q) = d\frac{rel(s|q)}{\sum_{z\in C} rel(z|q)} + (1-d)\sum_{v\in C}\frac{sim(s,v)}{\sum_{z\in C} sim(z,v)}p(v|q)$$

  - d controls ‘bias’: i.e. relative weighting

Tuning & Assessment

- Parameters:
  - Similarity threshold: filters adjacency matrix
  - Question bias: Weights emphasis on question focus
- Parameter sweep:
  - Best similarity threshold: 0.14-0.2
    - As before
  - Best question bias: high: 0.8-0.95
- Question bias in LexRank can improve

Other Strategies

- Methods depend on base system design
  - All aim to incorporate similarity with query/topic
- CLASSY HMM:
  - Add question overlap feature to HMM vector
    - Log (# query tokens in sentence + 1)
      - Query tokens: tagged as noun, verb, adj, adv, or proper nouns
  - Other, more aggressive approach detrimental
- FastSum: SVM regression on sentences
  - Adds topic title frequency feature:
    - Proportion of words in sent which appear in title
- Others: Require minimum number of topic words

Overview

- Many similar strategies:
  - Features, weighting, ranking: overlap based
- Actual evaluation impact:
  - Not necessarily very large (e.g. 0.003 ROUGE)
    - But can be useful
  - Aggressive approaches can have large negative impact
    - I.e. explicitly adding NER spans

Optimization Approaches to Reducing Redundancy

- DPP: Determinantal Point Processes (Kulesza &Taskar, '12)
  - Set models balancing information importance w/diversity
- ICSISumm: Uses Integer Linear Programming frame
  - Optimizes coverage of key bigrams weighted by doc freq
- OCCAMS_V
  - Uses LSA (Latent Semantic Analysis) to weight terms
  - Sentence selection via optimization problems:
    - Budgeted maximal coverage; knapsack

ICSISumm

- Key ideas:
  - Cast summarization as optimization problem
  - Identify important “concepts” to incorporate
  - Build best such summary
  - Implemented as integer linear programming

Integer Linear Programming

- Aka ILP
- An integer linear program specifies:
  - A single linear maximization term
  - Subject to linear equality/inequality constraints
  - Involving integer valued variables

Summarization as ILP

- Map summary requirements to ILP elements

Summarization as ILP

- Summary goal:
  - “best” summary
- Summary requirements:
  - Minimize redundancy
  - Within desired length

- Maximization term:

$$\sum_i w_i c_i$$

- Implicit:
- Length constraint:

$$\sum_j l_j s_j < L$$

- Coverage constraint:
  - Concept covered by sent

$$\sum_j s_j o_{ij} \ge c_i \forall i$$

$$s_j o_{ij} \le c_i \forall i, j$$

Representing Concepts

- Concepts = Bigrams
  - Stemmed
  - No stopword-only bigrams
  - Occurring in at least 3 documents
- Weights:
  - Document frequency:
  - # of documents (from cluster) for bigram
- Selected sentences must contain >= 2 query terms

Results

- After using open source solver

- 2009 results:
  - 2nd best pyramid, ROUGE-2
  - Best ROUGE-3, ROUGE-4

(Interesting sentence compression: later…)