Web Search and Text Mining

Lecture 25: Document Summarization

Outline:

- Generic extractive summarization
- The mutual reinforcement principle for saliency score computation
- Sentence clustering using linear order prior

Text Summarization

Text summarization is the process of automatically creating a compressed version of a given text that provides useful information for the user.

The information content of a summary depends on users needs.

Extractive summarization

> Extract text segments (sentences or paragraphs) to capture the essential content of documents

Generic summarization

General topics covered by a document (or a group of documents) vs. query-focused summarization exemplified by Google's snippets.

Generic summarization for the Web? Longer multi-topic documents (pdf and ps documents), potential use for hybrid system. User studies needed for investigating potential improved user experiences in search engines.

Basic ideas for extract generation

1. cluster sentences of a document (or a set of documents) into "topical groups"

2. within each topical groups select the keyphrases and sentences by their saliency scores.

Approaches

1. methods for exploring linear ordering of the sentences to enhance sentence clustering quality.

2. the mutual reinforcement principle for simultaneous keyphrase and sentence saliency score computation (adaptation of Hub-Authority to summarization).

The Mutual Reinforcement Principle

For a topical group, generating two sets of objects:

$$T = \{t_1, \ldots, t_n\}, \quad S = \{s_1, \ldots, s_m\}$$

the set of terms and the set of sentences.

Build a *bipartite* graph with T and S as the two sets of vertices. E.g., if the term t_i appears in sentence s_j, we then create an edge between t_i and s_j. $\Rightarrow G(T, S, W)$.

Potential features for computing edge weights

term frequency

term position

pos information

number of words in terms

sentence length

Computation of weight function

For a term-sentence pair (i, j), generate a list of features $f_1, \ldots, f_k$, the weight function

$$w(i, j) = W(f_1, \ldots, f_k).$$

Mutual reinforcement principle

Applicable in many situations, examples: 1) Hubs and authorities of Web pages, (Kleinberg, 1999). 2) Judges and items being judged.

Need to compute saliency scores $u(t_i)$ for term t_i and $v(s_j)$ for sentence s_j.

> A term should have a high saliency score if it appears in many sentences with high saliency scores while a sentence should have a high saliency score if it contains many terms with high saliency scores.

- the saliency score of a term is determined by the saliency scores of the sentences it appears in.

- the saliency score of a sentence is determined by the saliency scores of the terms it contains.

Mathematically, the previous statement can be written as

$$u(t_i) \propto \sum_{s_j \sim t_i} w_{ij} v(s_j),$$
$$v(s_j) \propto \sum_{t_i \sim s_i} w_{ij} u(t_i),$$

where $s_j \sim t_i$ indicates there is an edge between term t_i and sentence s_j. Equivalently in matrix format,

$$u = \frac{1}{\sigma} W v, \quad v = \frac{1}{\sigma} W^T u.$$

Choose $\sigma = \sigma_{\max}(W)$, the maximum singular value of W, it is guaranteed that both u and v have *nonnegative* components. The component values of u and v give the term and sentence saliency scores, respectively.

Computational issues

For computing the triplet $\{u, \sigma, v\}$, we can use a variation of the power method:

Start with an arbitrary v, alternate between the following two steps until convergence

1. Compute and normalize

$$u = Wv, \quad u = u/\|u\|,$$

2. Compute and normalize

$$v = W^T v, \quad v = v/\|v\|,$$

Convergence Theory: linear convergence at rate $\sigma_2(W)/\sigma_1(W)$.

Clustering sentence into topical groups

Existing methods tend to consider documents as a bag of sentences ignoring sentence ordering. Tow potential remedies

1. First apply text segmentation, then cluster text segments

2. Incorporating sentence ordering into sentence clustering

Using sentence ordering

> Documents consist of sentences arranged in a *linear* order, and near-by sentences in terms of this intrinsic linear order tend to be about the same topic.

Build an *undirected weighted* graph with vertices representing the sentences of a document and the *similarity* w_{ij} of two sentences s_i and s_j computed by a similarity function based a set of features $\Longrightarrow W_S = (w_{ij})$.

Strengthen the similarity weight between near-by sentences

$$\widehat{w}_{ij} = \begin{cases} w_{ij} + \alpha & \text{if } s_i \text{ and } s_j \text{ are near-by} \\ w_{ij} & \text{otherwise} \end{cases}$$

α the sentence link strength, and the modified weight matrix is denoted by $W_S(\alpha)$.

Spectral K-means for clustering (Z., Ding, Gu, He and Simon, NIPS 14, 2002)

Clustering based on objects similarity instead of object feature vectors.

The sum-of-squares clustering cost function for K-means, feature vector set $A = [a_1, \ldots, a_n]$, a set of means $[m_1, \ldots, m_k]$ associated with a partition.

$$J(A, M) = \sum_{i=1}^{n} \|a_i - m_{[i]}\|^2,$$

i.e., $m_{[i]}$ is the closest mean to a_i. Let s_i be the number of vectors in cluster i, set

$$X = \mathrm{diag}(e/\sqrt{s_1}, \ldots, e/\sqrt{s_k}),$$

we have

$$J(A, M) = \text{trace}(A^T A) - \text{trace}(X^T A^T A X),$$

and its minimization is equivalent to

$$\max\{\ \text{trace}(X^T A^T A X) \mid X = \text{diag}(e/\sqrt{s_1}, \ldots, e/\sqrt{s_k})\}.$$

Spectral relaxation,

$$\max\{\ \text{trace}(X^T A^T A X) \mid X \text{ orthonormal}\}.$$

Solution given by the first k largest eigenvectors of $A^T A$. Kernelization, $A^T A \Longrightarrow K = [K(a_i, a_j)]$.

Sentence clustering algorithm

- Compute the k eigenvectors $V_k = [v_1, \ldots, v_k]$ of $W_S(\alpha)$ corresponding to the largest k eigenvalues.

- Compute the pivoted QR decomposition of V_k^T as

$$(V_k)^T P = QR = Q[R_{11}, R_{12}],$$

  Q k-by-k orthogonal, R_{11} k-by-k upper triangular, and P a permutation matrix.

- Compute

$$\widehat{R} = R_{11}^{-1}[R_{11}, R_{12}]P^T = [I_k, R_{11}^{-1}R_{12}]P^T.$$

Then the cluster membership of each sentence is determined by the row index of the largest element *in absolute value* of the corresponding column of $\widehat{R}$.

Choosing α using generalized cross-validation (GCV)

- Apply spectral clustering to $W(\alpha)$ to obtain a set of sentence clusters $\Pi^*(\alpha)$

- Compute $\gamma(\Pi^*(\alpha))$, the number of consecutive sentence segments it generates

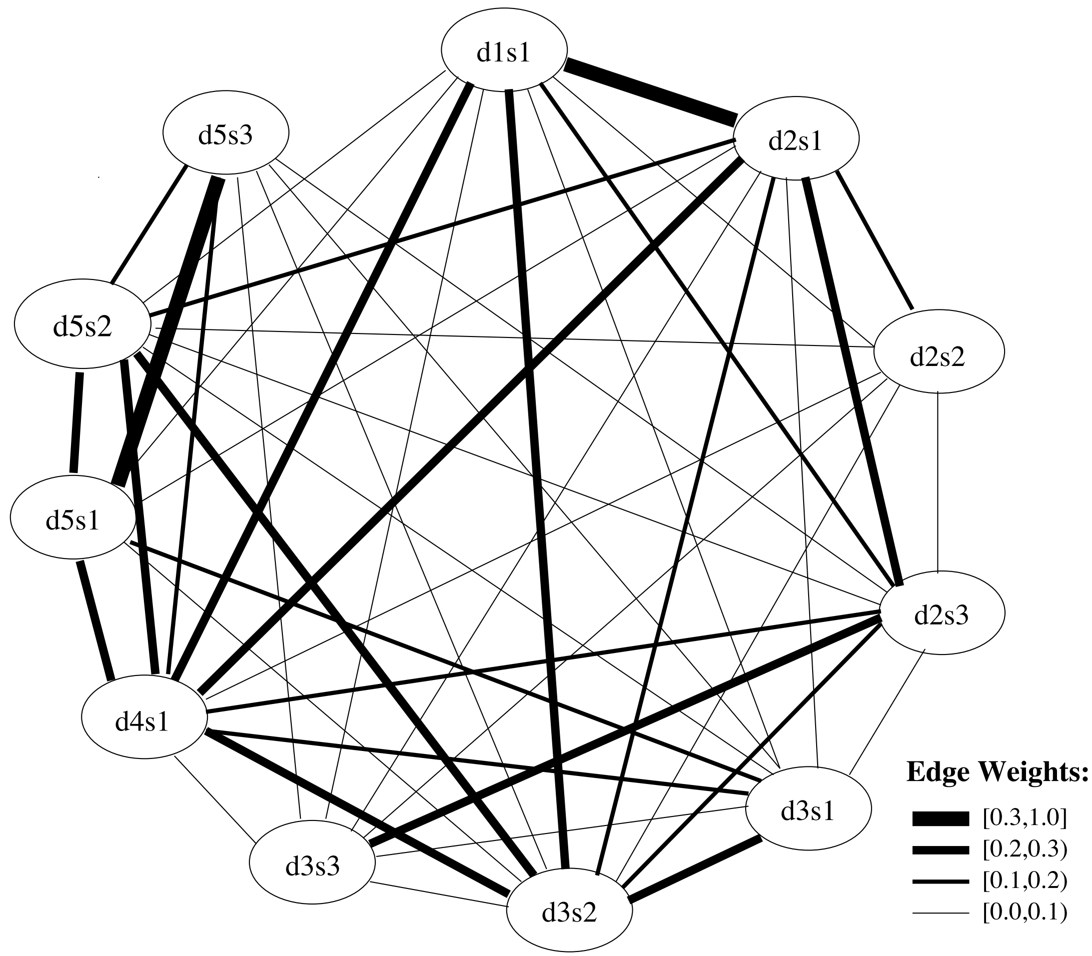
- Maximize

$$\text{GCV}(\alpha) = (n - k - J(W, \Pi^*(\alpha)))\,/\gamma(\Pi^*(\alpha)),$$

LexRank

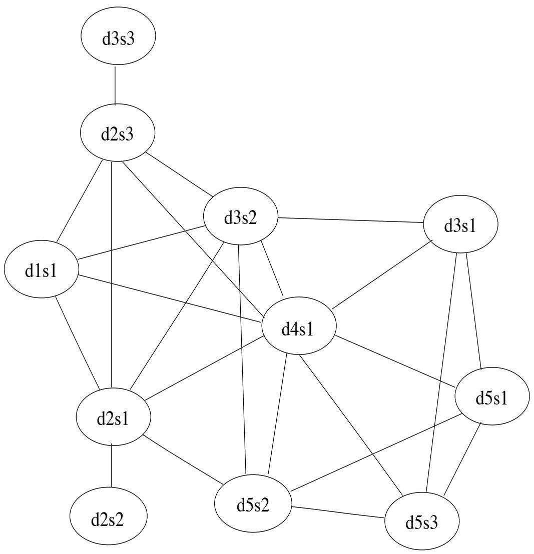
Examine sentence centrality for extractive summarization.

SNo	ID	Text
1	d1s1	Iraqi Vice President Taha Yassin Ramadan announced today, Sunday, that Iraq refuses to back down from its decision to stop cooperating with disarmament inspectors before its demands are met.
2	d2s1	Iraqi Vice president Taha Yassin Ramadan announced today, Thursday, that Iraq rejects cooperating with the United Nations except on the issue of lifting the blockade imposed upon it since the year 1990.
3	d2s2	Ramadan told reporters in Baghdad that "Iraq cannot deal positively with whoever represents the Security Council unless there was a clear stance on the issue of lifting the blockade off of it.
4	d2s3	Baghdad had decided late last October to completely cease cooperating with the inspectors of the United Nations Special Commission (UNSCOM), in charge of disarming Iraq's weapons, and whose work became very limited since the fifth of August, and announced it will not resume its cooperation with the Commission even if it were subjected to a military operation.
5	d3s1	The Russian Foreign Minister, Igor Ivanov, warned today, Wednesday against using force against Iraq, which will destroy, according to him, seven years of difficult diplomatic work and will complicate the regional situation in the area.


d1s1
d2s1
d5s3
d5s2
d2s2
d5s1
d2s3
d4s1
d3s1
d3s3
d3s2
Edge Weights:
[0.3,1.0]
[0.2,0.3)
[0.1,0.2)
[0.0,0.1)

Degree Centrality

ID	Degree (0.1)	Degree (0.2)	Degree (0.3)
d1s1	5	4	2
d2s1	7	4	2
d2s2	2	1	1
d2s3	6	3	1
d3s1	5	2	1
d3s2	7	5	1
d3s3	2	2	1
d4s1	9	6	1
d5s1	5	4	2
d5s2	6	4	1
d5s3	5	2	2



LexRank and PageRank

Let $p(u)$ be the centrality of sentence u, then

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