TOPICRANK: GRAPH-BASED TOPIC RANKING FOR KEYPHRASE EXTRACTION

reporter: Ning Li

Ning Li

Adrien Bougouin and **Florian Boudin** and **Béatrice Daille**
Université de Nantes, LINA, France
{adrien.bougouin,florian.boudin,beatrice.daille}@univ-nantes.fr

OUTLINE

- Introduction
- Topic Identification
- Graph-Based Ranking
- Keypharse Selection
- Experimental Settings
- Results
- Conclusion and Future Work

Introduction

- Keyphrase
  - A set of terms in a document that give a brief summary of its content for readers.
  - Used in information retrieval and digital library
  - An essential step in document categorization, clustering and summarization

- Problem
  - Most of documents have no associated keyphrases

→ Automatic keyphrase extraction

Introduction

Automatic keyphrase extraction

- Supervised method
  - Binary classification task
- Unsupervised method
  - Language modeling
  - Clustering
  - Graph-based ranking

Introduction

Graph-based ranking

- TextRank
- SingleRank
- TopicRank

Introduction

- TextRank method
  - Derived from PageRank
  - Represent a document by a graph where words are vertices and edges represent co-occurrence relations.
  - Assign a significance score to each word

Introduction

- TextRank method

程序员(英文Programmer)是从事程序开发、维护的专业人员。一般将程序员分为程序设计人员和程序编码人员，但两者的界限并不非常清楚，特别是在中国。软件从业人员分为初级程序员、高级程序员、系统分析员和项目经理四大类。

→

[程序员/n, (, 英文/nz, programmer/en,), 是/v, 从事/v, 程序/n, 开发/v, 、/w, 维护/v, 的/uj, 专业/n, 人员/n, 。/w, 一般/a, 将/d, 程序员/n, 分为/v, 程序/n, 设计/vn, 人员/n, 和/c, 程序/n, 编码/n, 人员/n, ，/w, 但/c, 两者/r, 的/uj, 界限/n, 并/c, 不/d, 非常/d, 清楚/a, ，/w, 特别/d, 是/v, 在/p, 中国/ns, 。/w, 软件/n, 从业/b, 人员/n, 分为/v, 初级/b, 程序员/n, 、/w, 高级/a, 程序员/n, 、/w, 系统/n, 分析员/n, 和/c, 项目/n, 经理/n, 四/m, 大/a, 类/q, 。/w]

→

[程序员, 英文, 程序, 开发, 维护, 专业, 人员, 程序员, 分为, 程序, 设计, 人员, 程序, 编码, 人员, 界限, 特别, 中国, 软件, 人员, 分为, 程序员, 高级, 程序员, 系统, 分析员, 项目, 经理]

Introduction

- SingleRank method
  - weights the edges with the number of co-occurrences
  - no longer extracts keyphrases by assembling ranked words

- Keyphrases
  - noun phrases extracted from the document
  - ranked according to the sum of the significance of the words they contain.

- Disadvantage
  - tend to assign high scores to long but non important phrases
    (“nash equilibrium”和”unique nash equilibrium”)

TopicRank

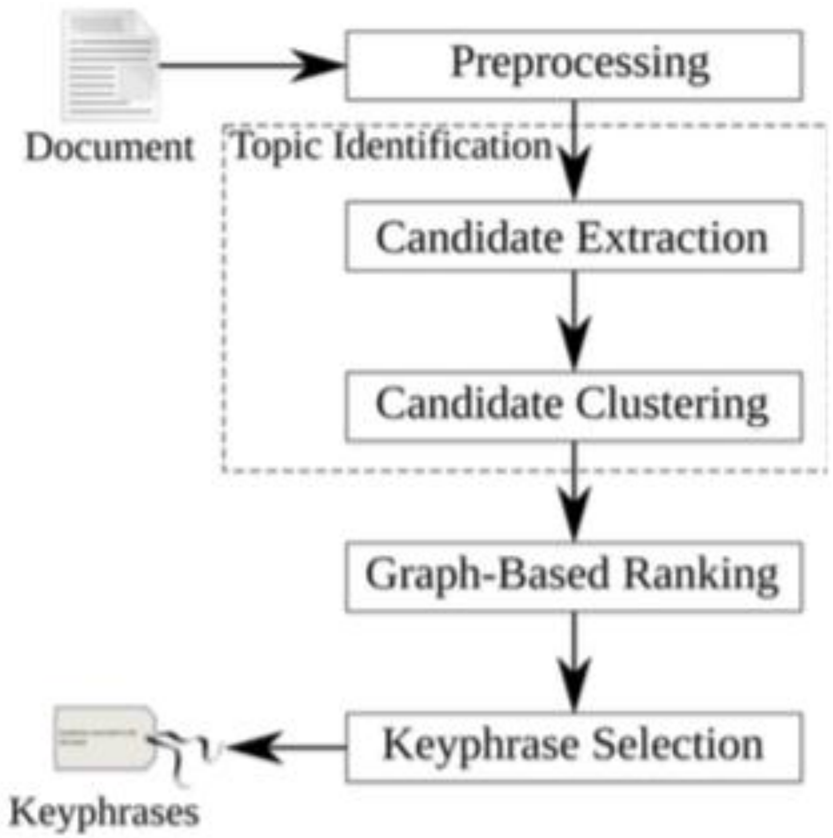


Figure 1: Processing steps of TopicRank.

Topic Identification

- Keyphrase candidates
  - extract the longest sequences of nouns and adjectives from the document

- Keyphrase candidates-> topic
- Hierarchical Agglomerative Clustering (HAC) algorithm
  - complete linkage

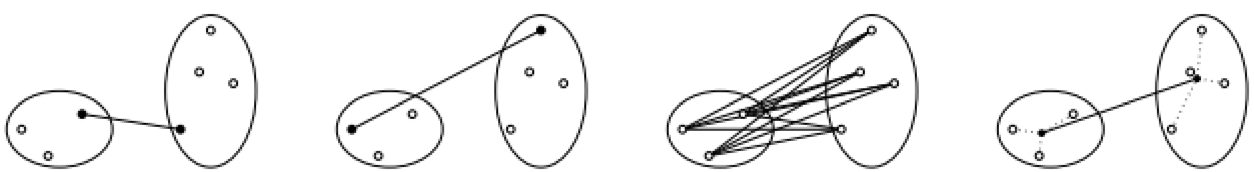
    more likely to group topically unrelated candidates
  - single linkage

    less likely to group topically related candidates
  - Average linkage
  - Centroid

Topic Identification

- (HAC) algorithm

  - Start with all objects in their own cluster
  - Repeat until there is only one cluster
  - Among the current clusters, determine the two clusters, c_i and c_j, that are closest Replace c_i and c_j with a single cluster $c_i \cup c_j$



(a) Single-Linkage (b) Complete-Linkage (c) Average-Linkage (d) Centroid

Graph-Based Ranking

- Graph Construction
  - G = (V; E) : a complete and undirected graph
  - V : a set of vertices and the edges
  - E : a subset of V×V .
  - t_i , t_j : two topics
  - c_i , c_j : the candidate keyphrases c_i, c_j belong to the topic
  - pos(c_j): all the offset positions of the candidate keyphrase c_j

$$w_{i,j} = \sum_{c_i \in t_i} \sum_{c_j \in t_j} \mathrm{dist}(c_i, c_j) \qquad (1)$$

$$\mathrm{dist}(c_i, c_j) = \sum_{p_i \in \mathrm{pos}(c_i)} \sum_{p_j \in \mathrm{pos}(c_j)} \frac{1}{|p_i - p_j|} \qquad (2)$$

Graph-Based Ranking

- Subject Ranking
  - G = (V; E) : a complete and undirected graph
  - V_j: the topics voting for t_i
  - λ: a damping factor generally defined to 0.85

$$S(t_i) = (1 - \lambda) + \lambda \times \sum_{t_j \in V_i} \frac{w_{j,i} \times S(t_j)}{\sum_{t_k \in V_j} w_{j,k}} \quad (3)$$

Keyphrase selection

- Advantage
  - avoid redundancy
  - lead to a good coverage of document topics
- Three Strategies
  - select candidate that appears first in the document.
  - select candidate that is most frequently used.
  - select the centroid of the cluster

Keyphrase selection

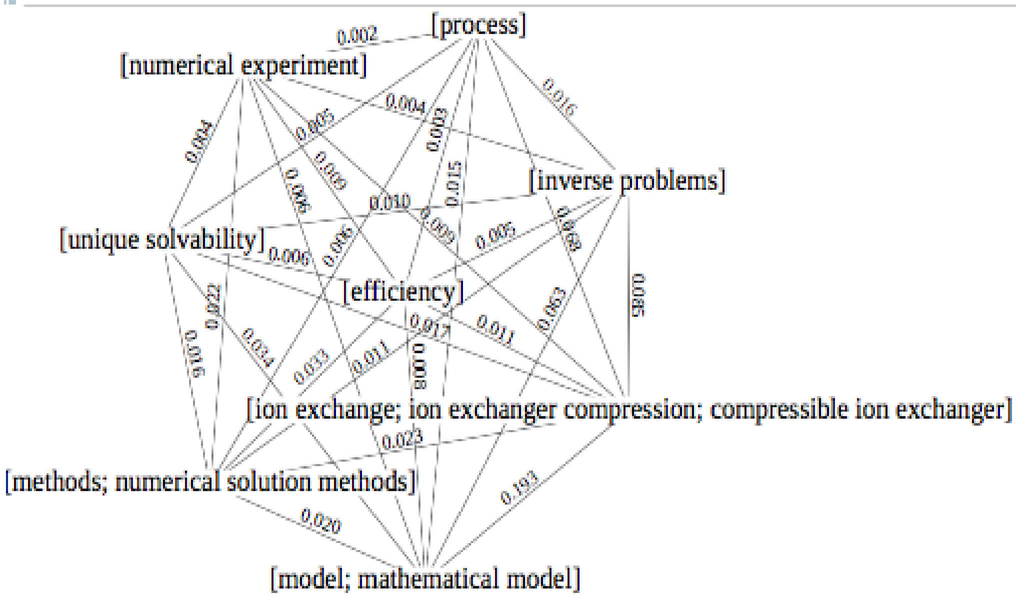


Figure 2: Sample graph build by TopicRank from Inspec, file *2040.abstr*.

Keyphrases assigned by human annotators:
ion exchange; mathematical model; inverse problems; numerical solution methods; unique solvability; compressible ion exchanger; ion exchanger compression
Keyphrases assigned by TopicRank:
ion exchange; ***mathematical model***; ***inverse problems***; ***numerical solution methods***; process; ***unique solvability***; efficiency; numerical experiment

Experimental settings

- Datasets

Corpus	Documents				Keyphrases		
	Type	Language	Number	Tokens average	Total	Average	Missing
Inspec	Abstracts	English	500	136.3	4913	9.8	21.8%
SemEval	Papers	English	100	5179.6	1466	14.7	19.3%
WikiNews	News	French	100	309.6	964	9.6	4.4%
DEFT	Papers	French	93	6844.0	485	5.2	18.2%

Table 1: Dataset statistics (missing keyphrases are counted based on their stemmed form).

Experimental settings

- Preprocessing
  - Sentence segmentation
  - Word tokenization
    - English： Tree bank Word Tokenizer
    - French： the Bonsai word tokenizer
  - Part-of-Speech tagging
    - English： Stanford POS- tagger and
    - French： MElt

Experimental settings

- Baselines
  - TextRank
  - SingleRank
  - TF-IDF

Experimental settings

TF-IDF

- Term frequency: the frequency that term *i* occurs in document *j*

$$\mathrm{tf_{i,j}} = \frac{n_{i,j}}{\sum_k n_{k,j}}$$

- Inverse document frequency: a measure of how much information the word provides, whether the term is common or rare across all documents

$$\mathrm{idf_i} = \log \frac{|D|}{|\{j : t_i \in d_j\}|}$$

- $$\text{tf-idf}_{i,j} = \text{tf}_{i,j} \times \text{idf}_i$$

Experimental settings

- Evaluation measures
  - Precision
  - Recall
  - F-score
    - $\text{F-score} = 2 \times \frac{Precision \times Recall}{Precision + Recall}$

Results

- The first experiment compares TopicRank to the baselines

Methods	Inspec			SemEval			WikiNews			DEFT		
	P	R	F	P	R	F	P	R	F	P	R	F
TF-IDF	32.7	38.6	33.4	13.2	8.9	10.5	33.9	35.9	34.3	10.3	19.1	13.2
TextRank	14.2	12.5	12.7	7.9	4.5	5.6	9.3	8.3	8.6	4.9	7.1	5.7
SingleRank	34.8	40.4	**35.2**	4.6	3.2	3.7	19.4	20.7	19.7	4.5	9.0	5.9
TopicRank	27.6	31.5	27.9	14.9	10.3	**12.1**†	35.0	37.5	**35.6**†	11.7	21.7	**15.1**†

Table 2: Comparison of TF-IDF, TextRank, SingleRank and TopicRank methods, when extracting a maximum of 10 keyphrases. Results are expressed as a percentage of precision (P), recall (R) and f-score (F). † indicates TopicRank's significant improvement over TextRank and SingleRank at 0.001 level using Student's t-test.

Results

- The second experiment individually evaluates the modifications of topicRank compared to singleRank

Methods	Inspec			SemEval			WikiNews			DEFT		
	P	R	F	P	R	F	P	R	F	P	R	F
SingleRank	34.8	40.4	35.2	4.6	3.2	3.7	19.4	20.7	19.7	4.5	9.0	5.9
+phrases	21.5	25.9	22.1	9.6	7.0	8.0†	28.6	30.1	28.9†	10.5	19.7	13.5†
+topics	26.6	30.2	26.8	14.7	10.2	11.9†	31.0	32.8	31.4†	11.5	21.4	14.8†
+complete	34.9	41.0	**35.5**	5.5	3.8	4.4	20.0	21.4	20.3	4.4	9.0	5.8
TopicRank	27.6	31.5	27.9	14.9	10.3	**12.1**†	35.0	37.5	**35.6**†	11.7	21.7	**15.1**†

Table 3: Comparison of the individual modifications from SingleRank to TopicRank, when extracting a maximum of 10 keyphrases. Results are expressed as a percentage of precision (P), recall (R) and f-score (F). † indicates a significant improvement over SingleRank at 0.001 level using Student's t-test.

- SingleRank's viertice: keyphrase candidates (+phrases), topics (+topics)
- TopicRank's vertice: word vertices (+complete)

Results

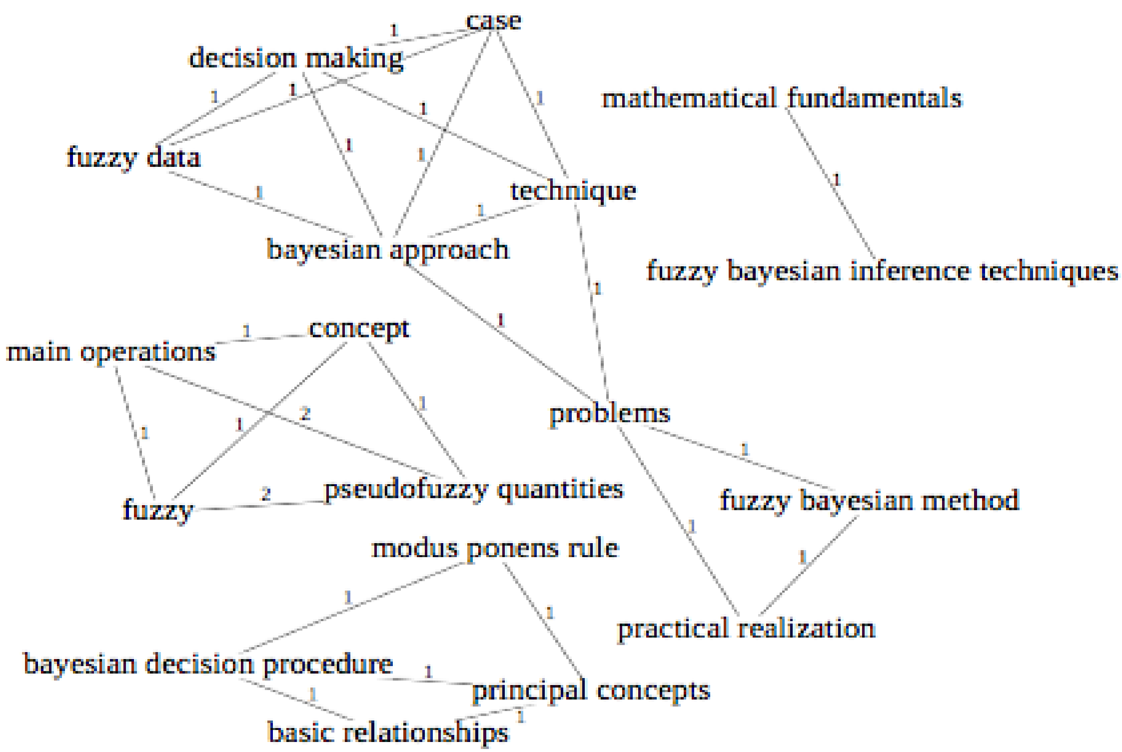


Figure 3: Connected component problem with the method SingleRank+phrases. Example taken from Inspec, file *1931.abstr*.

Results

- The last experiment compares the keyphrase selection strategies

Methods	Inspec			SemEval			WikiNews			DEFT		
	P	R	F	P	R	F	P	R	F	P	R	F
First position	27.6	31.5	27.9	14.9	10.3	12.1†	35.0	37.5	35.6†	11.7	21.7	15.1†
Frequency	26.7	30.2	26.8	1.7	1.2	1.4	25.7	27.6	26.2	1.9	3.8	2.5
Centroid	24.5	28.0	24.7	1.9	1.2	1.5	28.1	29.9	28.5	2.6	5.0	3.4
Upper bound	36.4	39.0	**35.6**	37.6	25.8	**30.3**	42.5	44.8	**42.9**	14.9	28.0	**19.3**

Table 4: Comparison of the keyphrase candidate selection strategies against the best possible strategy (upper bound), when extracting a maximum of 10 keyphrases. Results are expressed as a percentage of precision (P), recall (R) and f-score (F). † indicates the first position strategy's significant improvement over the frequency and the centroid strategies at 0.001 level using Student's t-test.

Upper bound: compute the result the number of correct matches is equal to the number of clusters containing at least one reference keyphrase.

Conclusion and future work

- Conclusion
  - More straightforward way to identify the set of keyphrases that covers the main topics of a document.
  - Eliminate redundancy while reinforcing edges
  - Use of a complete graph that better captures the semantic relations between topics

Conclusion and future work

- Future work
  - Improve the topic identification and the keyphrase selection
  - Develop an evaluation process to determine cluster quality
  - Investigate the use of linguistic knowledge for similarity measures

Q&A