Overview of Automatic Keyphrase Extraction

Keyphrase Extraction

Su Nam Kim

Search Engine Technology
at RMIT

August 3, 2010

Content

What do we study today?

- What is **Keyphrases**?
- Why do we need to get them?
- How do we get them **automatically**?
- How do we validate extracted keyphrases?

Overview (I)

What is a term???

Overview (I)

Definition of Term: single or group of words that express semantics (e.g. *apple, fast, perform, frequently* vs. a, of, the)

The study related to terms :

- Term Extraction
  - technical terms: noun, verb, adjective
  - keyphrases: noun, noun phrase
  - synonym/hypernym/sister words/etc. : e.g. *plant*:industrial plant/building complex
- Term Categorization : categorize terms with a set of predefined classes:
  - domain-specific terms: Dell, HTML, Web in *Computing*
  - word sense disambiguation: w/ *plant* industrial plant vs. flora

Overview (II)

Today's Topic automatically extract keyphrases

Keyphrases words which represent the topic of articles

Example:

Title: New lower bounds of the size of error-correcting codes for the Z-channel

Content: Optimization problems on graphs are formulated to obtain new lower bounds of the size of error-correcting codes for the Z-channel

Overview (II)

Today's Topic automatically extract keyphrases
Keyphrases words which represent the topic of articles
Example: Document 1
Title: New lower bounds of the size of error-correcting codes for the Z-channel
Content: Optimization problems on graphs are formulated to obtain new lower bounds of the size of error-correcting codes for the Z-channel
Keywords: error-correction code, Z-channel, optimatization

Significance (I)

used for many NLP applications

- semantic metadata for summarization (Barzilay:1997, Lawrie:2001, DAvanzo:2005)
- document indexing (Gutwin:1999)
- document clustering (Zhang:2004, Hammouda:2005)
- document summarization (Berger:2000, Buyukkokten:2001)

Example Data:
Doc1 = error-correction codes, lower bounds, Z-channel, optimatization problem
Doc2 = Z-channel, optimatization problems
Doc3 = error-correction codes, algorithm, efficiency
Doc4 = Z-channel, error-correction codes, optimatization
Doc5 = algorithm, efficiency, graph

Significance (II)

- semantic metadata/short summary: keyphrases are used as seed to generated summary.
  e.g. Doc5: *algorithm* improve the *efficiency* of searching *graph*
- document indexing (Gutwin:1999)
  e.g. w/ a query word, *Z-channel*
- document clustering (Zhang:2004, Hammouda:2005)

Significance (II)

- semantic metadata/short summary: keyphrases are used as seed to generated summary.
  e.g. Doc5: *algorithm* improve the *efficiency* of searching *graph*
- document indexing
  e.g. w/ a query word, *Z-channel*, *Doc1, Doc2, Doc4* is now considered as relevant documents
- document clustering

Significance (II)

- semantic metadata/short summary: keyphrases are used as seed to generated summary.
  e.g. Doc5: *algorithm* improve the *efficiency* of searching *graph*
- document indexing
  e.g. w/ a query word, *Z-channel*, *Doc1, Doc2, Doc4* is now considered as relevant documents
- document clustering
  e.g. Cluster_1(Doc1, Doc2, Doc4), Cluster_2(Doc3, Doc5)

Difficulties (I)

Process Preprocessing → Candidate Extraction → Candidate Ranking (→ Evaluation)

- identify term vs. non-term (*candidate selection*) → NN, NP e.g. *distributed computing, grid algorithm, ad-hoc, agent's price, demand and price, efficiency of search*
- dealing with variations (*candidate selection*)
- specification vs. generalization (*ranking candidates*)

Difficulties (I)

Process Preprocessing → Candidate Extraction → Candidate Ranking (→ Evaluation)

- identify term vs. non-term (*candidate selection*) → NN, NP e.g. *distributed computing, grid algorithm, agent's price, demand and price, efficiency of search* vs. ad-hoc
- dealing with variations (*candidate selection*) → plurality (*algorithm* vs. *algorithms*), possessive (*agent's price* vs. *agent price*), preposition (*effiency of search* vs. *search effiency*), conjunction (*demand and price*)
- specification vs. generalization (*ranking candidates*) e.g. algorithm, generalization, computing, search

Related Work (I)

- KEA(Frank:1999, Witten:1999, Medelyan:2006): TF·IDF and first occurrance of word, domain specific (index as candidates)
- GenEx (Turney:1999, 2000) : 9 different syntactic features such as length, frequency of stem etc., decision tree induction
- Fung:1998 : automatic keyphrase extraction in Chinese and Japanese
- Textract (Park:2004) : domain-specific cohesion (Damerau:1993) & term cohesion (Dice:1945)
- Barker:2000 : using length, frequency & head noun frequency

Related Work (II)

- Tomokiyo:2003 : using information loss between foreground & background data based on 1 vs. n-gram models
- Turney:2003 : Keyphrase cohesion (among top N and the remaining, check keyphrase cohesion)
- Mihalcea:2004 : *TextRank*, graph-based, simplex terms using frequency of co-occurances
- Nguyen:2007 : using lingistic features such as section, POS sequence
- Wan:2008 : referring clustered documents as a domain info.
- Liu:2009 : using clustering algorithm, pick up terms close to the centroids. unsupervised.

Nature of Keyphrase

- form: simplex nouns or noun phrases (NPs)
- NPs as keyphrases : nouns with adjective(s), occasionally adverbs or other POSs (e.g. *dynamically allocated task*)
- can contain hypens (e.g. *sensor-grouping, multi-agent system*) and apostrophes (e.g. *Bayes' theroem, agent's goal*)
- length observation: few 3-term noun sequences are longer than 3-term NPs (Paukkeri:2008)
- many form with a preposition (e.g. *quality <u>of</u> service, incentive <u>for</u> cooperation*)
- few form in conjunction form (e.g. *behavioral and evolution and extrapolation*)
- can occur as abbreviations (e.g. *POMDP = partially observable Markov decision process*)

Keyphrase Variation

(cf. similar to Machine Translation)

- *word order* fixed (e.g. *service quality* $\neq$ *quality service*)
- *word adjacency* fixed (e.g. *quality serivce* $\neq$ *quality ... service*)
- *morphological* variation allowed (e.g. *quality/qualities/...*)
- *lexical semanitcs* allowed but high cost to check (e.g. *multiagent behavior = multiagent action/manner*)
- *string overlap* allowed (e.g. *grid computing = grid computing algorithm*)

Candidate Selection: Approaches

- Issues: form, variation, length, frequency
- KEA uses the index words as candidates (Food & Agriculture domain)
- GenEx uses $1-3$ sequence words (i.e. n-grams)
- Textract uses regular expressions to extract noun sequences
- Nguyen & Kan uses regular expressions to extract both noun sequences and simple NP w/ preposition, *of* (i.e. *NN of NN*)

Candidate Selection (Kim et. al. 2009)

Table: Candidate Selection Rules

Criteria	Rule
Frequency	(**Rule1**) *Frequency heuristic* i.e. frequency $\geq$ 2 for simplex words vs. frequency $\geq$ 1 for NPs
Length	(**Rule2**) *Length heuristic* i.e. up to length 3 for NPs in non-*of-PP form* vs. up to length 4 for NPs in *of-PP form* (e.g. *synchronous concurrent program vs. model of multiagent interaction*)
Alternation	(**Rule3**) *of-PP form alternation* (e.g. *number of sensor = sensor number, history of past encounter = past encounter history*) (**Rule4**) *Possessive alternation* (e.g. *agent's goal = goal of agent, security's value = value of security*)
Extraction	(**Rule5**) *Noun Phrase* = $(NN\|NNS\|NNP\|NNPS\|JJ\|JJR\|JJS)^*(NN\|NNS\|NNP\|NNPS)$ (e.g. *complexity, effective algorithm, grid computing, distributed web-service discovery architecture*) (**Rule6**) *Noun Phrase <u>IN</u> Noun Phrase* (e.g. *quality of service, sensitivity of VOIP traffic, **VOIP traffic**,* *simplified instantiation of zebroid, **simplified instantiation***)

Candidate Selection : Exercise (I)

Title: New lower bounds of the size of error-correcting codes for the Z-channel
Content: Optimization problems on graphs are formulated to obtain new lower bounds of the size of error-correcting codes for the Z-channel
Title: New/JJ lower/JJR bounds/NNS of/IN the/DT size/NN of/IN error-correcting/JJ codes/NNS for/IN the/DT Z-channel/NNP
Content: Optimization/NN problems/NNS on/IN graphs/NNS are/VBP formulated/VBN to/TO obtain/VB new/JJ lower/JJR bounds/NNS of/IN the/DT size/NN of/IN error-correcting/JJ codes/NNS for/IN the/DT Z-channel/NNP

- Method1: unigrams and bigrams only
- Method2: POS pattern (NN, NN/NN, JJ/NN):
  NN(S):noun(s),NNP(S):proper noun(s),JJ(X):adjective,VB(X):verb,IN:preposition,DT:determiner

Candidate Selection : Exercise (II)

Method1: 1,2-grams

- *1-gram*: *New,lower,bounds*,of,the,*size*,of,*error-correcting,codes*,for,the,*Z-channel,Optimization,problems*,on,*graphs*,are,*formulated*,to,*obta*
  correcting,codes,for,the,*Z-channel*
- *2-gram*: *New lower,lower bounds*,bounds of,of the,the size,size of,of error-correcting,*error-correcting codes*,codes for,for the,the Z-channel,*Optimization problems*,problems on,on graphs,graphs are,are formulated,formulated to,to obtain,*obtain new,new lower,lower bounds*,bounds of,of the,the size,size of,of error-correcting,*error-correcting codes*,codes for,for the,the Z-channel

Candidate Selection : Exercise (III)

Method2: POS pattern: *lower/JJR bounds/NNS, size/NN, error-correcting/JJ codes/NNS, Z-channel/NNP, Optimatization/NN problems/NNS, graphs/NNS, lower/JJR bounds/NNS, size/NN, error-correcting/JJ codes/NNS, Z-channel/NNP*

Feature Selection

- Document Cohesion How likely keyphrases correlated with the document
- Keyphrase Cohesion among keyphrases, they share the same or similar semantics
- Term Cohesion component asociation is high if the components are a term/keyphrases (Church & Hanks 1989)
- Features not belong to above e.g. POS sequence(Hulth 2006), Suffix sequence, Acronym

Document Cohesion (I)

- TF * IDF (Frank et al. 1999, Witen et al. 1999) indicates the correlation between keyphrases and document
  - variation: different TF weighting for simplex vs. NP candidates
  - problem : IDF is not easy to measure due to the data size
  - observation1: using web data (Google n-gram), IDF is similar
  - observation2: substring can be counted as TF (e.g. *grid computing* for *effective grid computing, grid computing algorithm*)

Document Cohesion (II)

- w/ Context words (Kim & Wilbur 2001, Matsuo & Ishizuka 2003) using the context words to represent the corelation of candidates w.r.t. the document
  - idea came from term vs. non-term extraction
  - using context words (noun, verb, adjective) of candidates, represent candiates and measure the similarity between candidates and document
  - can be unsupervised approach

Keyphase Cohesion

- Keyphrase Cohesion (Turney 2003) assume that keyphrases have similar semantics. take top N and compute similarity between N and remaining candidates
- Co-occurrance of Candidate in Section assume that keyphrases appear in multiple key sections
- Title co-occurrence assume that title represetns topic of document and if keyphrases are in title, they have same/similar semantics

Keyphase Cohesion

- Keyphrase Cohesion (Turney 2003) assume that keyphrases have similar semantics. take top N and compute similarity between N and remaining candidates e.g.*top* N_{th} *candidates: cross-language information retrieval,query translation*
  other candidates: query suggestion,query analysis,query expansion,performance,machine translation
- Co-occurrance of Candidate in Section assume that keyphrases appear in multiple key sections
- Title co-occurrence assume that title represetns topic of document and if keyphrases are in title, they have same/similar semantics

Keyphase Cohesion

- Keyphrase Cohesion (Turney 2003) assume that keyphrases have similar semantics. take top N and compute similarity between N and remaining candidates e.g. *top N_{th} candidates: cross-language information retrieval,query translation*
  other candidates: query suggestion,query analysis,query expansion,*performance,machine translation*
- Co-occurrance of Candidate in Section assume that keyphrases appear in multiple key sections
- Title co-occurrence assume that title represetns topic of document and if keyphrases are in title, they have same/similar semantics

Term Cohesion

- this feature is from term vs. non-term extraction
- the component association is high if the components are term/keyphrases (Church:1945)
- adopted by Textract (Park:2004)
- usually applied to identify multi-term words

Term Cohesion

- this feature is from term vs. non-term extraction
- the component association is high if the components are term/keyphrases (Church:1945)
- adopted by Textract (Park:2004)
- usually applied to identify multi-term words
  e.g. *temporal text index,time-travel text search,web archive,ad-hoc search,information ad-hoc*

Term Cohesion

- this feature is from term vs. non-term extraction
- the component association is high if the components are term/keyphrases (Church:1945)
- adopted by Textract (Park:2004)
- usually applied to identify multi-term words
  e.g. *temporal text index,time-travel text search,web archive,ad-hoc search*,*information ad-hoc*

Other Features (I)

- First Appearance (KEA) assume that keyphrases occur in the begining (e.g. abstract, introduction)
- Section Information (Nguyen:2007) assume that keyphrases occur in key sections (abstract, introduction, related work, conclusion, title, reference, section head)
- Last Appearance

Other Features (II)

- POS sequence (Hulth:2006) e.g. *NN_NN, JJ_NN*
- Suffix sequence (Nguyen:2007) domain specific e.g. *refinement, information, maintemance*
- Acronym (Nguyen:2007) e.g. *POMDP=partially observable markov decision process*
- Length of Keyphrases (Barker:2000) assume that majority keyphrases are 2-3 term NSs

Feature Selection : Exercise (I)

Title: New (lower bounds) of the size of (error-correcting codes) for the Z-channel
Content: (Optimization problems) on graphs are formulated to obtain new (lower bounds) of the size of (error-correcting codes) for the Z-channel
Candidates from Method2: *lower bounds, size, error-correcting codes, Z-channel, Optimatization problems, graphs, lower bounds, size, error-correcting codes, Z-channel*

- TFXIDF
  - TF = $\frac{term\ freqency}{total\ No.\ of\ terms}$
  - IDF = $log(\frac{total\ No.\ of\ Docs}{No.\ of\ Docs\ containg\ target})$
- first appearance of words represented as *relative position*

Feature Selection : Exercise (II)

No. of words=27, Total No. of Docs=100

Table: Candidates with Features

Candidate	TFreq.	DFreq.	TFXIDF	Position
lower bounds	2	20	$\frac{2}{27}\log\frac{100}{20}$	$\frac{2}{27}$
size	2	60	$\frac{X1}{X2}\log\frac{X3}{X4}$	$\frac{X5}{X6}$
error-correcting codes	2	5	$\frac{X1}{X2}\log\frac{X3}{X4}$	$\frac{X5}{X6}$
Z-channel	2	2	$\frac{X1}{X2}\log\frac{X3}{X4}$	$\frac{X5}{X6}$
Optimatization problems	1	10	$\frac{X1}{X2}\log\frac{X3}{X4}$	$\frac{X5}{X6}$
graphs	1	30	$\frac{X1}{X2}\log\frac{X3}{X4}$	$\frac{X5}{X6}$

Feature Selection : Exercise (II)

No. of words=27, Total No. of Docs=100

Table: Candidates with Feature Values

Candidate	TFreq.	DFreq.	TFXIDF	Position
lower bounds	2	20	$\frac{2}{27}\log\frac{100}{20}$	$\frac{2}{27}$
size	2	60	$\frac{2}{27}\log\frac{100}{60}$	$\frac{5}{27}$
error-correcting codes	2	5	$\frac{2}{27}\log\frac{100}{5}$	$\frac{7}{27}$
Z-channel	2	2	$\frac{2}{27}\log\frac{100}{2}$	$\frac{10}{27}$
Optimatization problems	1	1	$\frac{1}{27}\log\frac{100}{1}$	$\frac{11}{27}$
graphs	1	30	$\frac{1}{27}\log\frac{100}{30}$	$\frac{13}{27}$

Feature Selection : Exercise (III)

Table: Candidates with Final Score

Ranking	Candidate	TFXIDF	Position	Score
1	Z-channel	0.29	0.63	0.18
2	error-correcting codes	0.22	0.74	0.16
3	lower bounds	0.12	0.93	0.11
4	Optimatization problems	0.17	0.59	0.10
5	size	0.04	0.81	0.03
6	graphs	0.04	0.52	0.02

top 5 Keyphrases: *Z-channel, error-correcting codes, lower bounds, Optimatization problems, size*

Evaluation Method

- Matching Number number of matching keywords in top 5, 10, 15 with precision, recall and f-score
  - partial matching doesn't receive credits
  - very limited variation of keyphrases (e.g. A of B $->$ B A)
- Semantic Similarity (Jamasz:2004)
  - using terabyte corpus to measure the Top candidates and keyphrases
  - require large corpus to measure it
- Domain Specific Thesaurus (Medelyan:2006)
  - using Agrovoc (food & agriculture), check similar words
- Wikipedia InterLink (Paukkeri:2008)
  - using the interlink among the multilingual documents
- R-precision & Weighted R-precision (Zesch:2009, Kim:2010)
  - $\frac{No.\ of\ matching\ words}{No.\ of\ longest\ phrases}$
  - applying weights per each component w.r.t component position
  - e.g. *distributed computing algorithm* vs. *computing algorithm*: R-precision = $\frac{2}{3}$, weighted R-precision = $\frac{\frac{2}{3}+\frac{3}{3}}{3}$

Evaluation : Exercise

$$Precision = \frac{No.\ of\ overlaps}{No.\ of\ assigned\ keyphrase}$$

$$Recall = \frac{No.\ of\ overlaps}{No.\ of\ gold-standard\ keyphrase}$$

$$F\text{-}score = 2 \text{ X } \frac{Precision\ X\ Recall}{Precison + Recall}$$

Gold-standard Keywords: *error-correction code, Z-channel, optimatization*

Extracted Keyphrases: *Z-channel, error-correcting codes, lower bounds, Optimatization problems, size*

- How many are matching before/after stemming= ?
- Precision = ?
- Recall = ?
- F-score = ?

Evaluation : Exercise

$$Precision = \frac{No.\ of\ overlaps}{No.\ of\ assigned\ keyphrase}$$
$$Recall = \frac{No.\ of\ overlaps}{No.\ of\ gold-standard\ keyphrase}$$
$$F\text{-}score = 2\ X\ \frac{Precision\ X\ Recall}{Precison + Recall}$$

Gold-standard Keywords: *error-correction code, Z-channel, optimatization*

Extracted Keyphrases: *Z-channel, error-correcting codes, lower bounds, Optimatization problems, size*

- How many are matching = exactly matching:*1* vs. matching after stemming:*2*
- Precision = ?
- Recall = ?
- F-score = ?

Evaluation : Exercise

$$Precision = \frac{No.\ of\ overlaps}{No.\ of\ assigned\ keyphrase}$$
$$Recall = \frac{No.\ of\ overlaps}{No.\ of\ gold-standard\ keyphrase}$$
$$F\text{-}score = 2\ \mathrm{X}\ \frac{Precision\ X\ Recall}{Precison + Recall}$$

Gold-standard Keywords: *error-correction code, Z-channel, optimatization*

Extracted Keyphrases: *Z-channel, error-correcting codes, lower bounds, Optimatization problems, size*

- How many are matching = matching after stemming:*2*
- Precision = $\frac{2}{5}$ = *0.40*
- Recall = $\frac{2}{3}$ = *0.67*
- F-score = 2 X $\frac{0.40\ X\ 0.67}{0.40+0.67}$ = *0.50*

SemEval 2010

- Goal1 : offer an opportunity to compete the systems $\rightarrow$ finding out the current status of task
- Goal2 : generate a standard data set for further research
  - 2,000 abstracts of journals in Inspec (Hulth:2003)
  - 120 documents in scientific articles with author & reader assigned keywords (Nguyen:2007)
  - 308 documents from DUC 2001 (Wan:2008)
- test, training & trial data contain 100, 144, 40 documents
- 20 teams participated and up to 3 submissions are allowed
- The best performances: 23.82%, 27.77% and 31.15% over author, reader, combined keyphrase sets
- For descriptions of the systems : check out workshop website *http://semeval2.fbk.eu/semeval2.php*
- data is available on request

Concluding Remarks

- explore three issues (i.e. **candidate selection, feature selection, evaluation metric**)
- **candidate selection** subject to the performances directly
- **feature selection** supervised vs. unsupervised, needs to be explored, especially for unsupervised automatic keyphrase extraction
- **evaluation metric** needs to be studied for general evaluation
- steady study but need to be improved for NLP applications
- Task 5 At Workshop on Semantic Evaluation (SemEval 2010) provided an opportunity to remark the current status of the task (data available on request)
- Publicly available datasets: listed on my website (www.csse.unimelb.edu.au/ snkim/research.html/)

Thank you

Any questions?