Selecting Candidates for Automatic Keyphrase Extraction

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

Abstract

Keyphrase extraction is the task of identifying single or multi-word expressions that best represent the content of a document. Most automatic keyphrase extraction methods rely on a candidate selection step, in which only the textual units that have similar properties to keyphrases are kept. Candidate selection therefore plays an important role in keyphrase extraction since it determines the upper bound of recall performance. In this paper, we compare three commonly used methods for candidate selection and propose a new approach that generates refined noun phrases by filtering out irrelevant adjective modifiers. Through experiments carried out on three standard datasets of different languages and domains, we show that our approach reduces the number of selected candidates and improves keyphrase extraction performance when used with either supervised or unsupervised methods.

1 Introduction

Since the last decade, the amount of information available on the web is constantly increasing. While the number of documents continues to grow, the need for efficient information retrieval methods becomes increasingly important. Keyphrases are single or multi-word expressions that represent the main content of a document. As they describe the main topics in documents, keyphrases are useful for many tasks such as information retrieval (Turney, 2000), summarization (D'Avanzo and Magnini, 2005) or document indexing (Medelyan and Witten, 2008). Despite this, only a small number of documents have keyphrases associated with them. Therefore, the automatic keyphrase extraction task has attracted a lot of attention (Kim et al., 2010).

The automatic keyphrase extraction task consists in the extraction of the most important textual units of a document. We distinguish two categories of keyphrase extraction methods: supervised methods and unsupervised methods. The first category of methods recasts the keyphrase extraction task as a binary classification task (Witten et al., 1999), whereas the second category often considers the keyphrase extraction task as a ranking task (Hasan and Ng, 2010). Although they handle the keyphrase extraction problem in their own different ways, supervised and unsupervised methods rely on the same preliminary steps. First, documents are preprocessed (word tokens, sentence boundaries, Part-of-Speech – POS, etc.). Second, a set of keyphrase candidates is extracted from the documents. Keyphrase candidates are textual units that can be extracted as keyphrases; Keyphrase candidates must have properties known to be properties of human-assigned keyphrases.

The selection of the keyphrase candidates (referred as candidate selection in the reset of this paper) is one of the most critical steps of the keyphrase extraction. The nature of the candidates must be carefully defined to suit the properties of human-assigned keyphrases and, most importantly, to avoid as much irrelevant candidates as possible. The range of proposed methods to select candidates is limitted. Commonly use methods select n-grams, NP-chunks or word sequences matching given patterns (Hulth, 2003), while Kim et al. (2009) use a more sofisticated method that filters candidates according to statistics learnt from training data. Based on infered keyphrase properties (Section 2), we argue the suitability of the commonly used methods and propose a new method that selects noun phrases and filters their irrelevant adjective modifiers (Section 3). Using three keyphrase extraction methods (Section 4), we analyse the quality of the candidate sets provide by each candidate selection methods and observe their impact on keyphrase extraction (Section 5). Results show that the quality of the candidate sets produced by our method is better and induce a higher performance on each keyphrase extraction method.

2 What is a Keyphrase?

In this section, we determine the properties of a keyphrase. First, we select standard keyphrase extraction datasets. Second, we extract statistics and infer keyphrase properties from their training sets.

2.1 Datasets

Keyphrase extraction datasets are collections of documents paired with reference keyphrases given by authors, readers or both. In our work, we use three standard datasets that differ in terms of document size, type and language.

The **DUC** dataset (Over, 2001) is a collection of 308 English news articles covering about 30 topics (e.g. tornadoes, gun control, etc.). This collection is the test dataset of the DUC-2001 summarization evaluation campaign and contains reference keyphrases annotated by Wan and Xiao (2008). We split the collection into two sets: a training set containing 208 documents and a test set containing 100 documents.

The **SemEval** dataset (Kim et al., 2010) contains 244 English scientific papers collected from the ACM Digital Libraries (conference and workshop papers). The papers are divided into two sets: a training set containing 144 documents and a test set containing 100 documents. The associated keyphrases are provided by both authors and readers.

The **DEFT** dataset (Paroubek et al., 2012) is a collection of 234 French scientific papers belonging to the *Humanities and Social Sciences* domain. DEFT is divided into two sets: a training set containing 141 documents and a test set containing 93 documents. Keyphrases provided with the documents of DEFT are given by authors.

2.2 Analysis of Reference Keyphrases

In this section, we aim to find general keyphrase properties from the analysis of the training sets of the previously mentioned datasets.

Table 1 shows statistics about the datasets and the keyphrases associated to their documents. First, keyphrases are presented regarding their number of words. Second, the multi-word keyphrases are presented regarding the Part-of-Speech of their words[1]. To obtain these Part-of-Speech, we automatically POS tagged the keyphrases of the English datasets with the Stanford POS tagger (Toutanova et al., 2003) and the keyphrases of the French dataset with MElt (Denis and Sagot, 2009). To avoid tagging errors, POS tagged keyphrases were manually corrected.

First, we observe (and confirm previous work observations) that most keyphrases are unigrams or bigrams ($\simeq$ 80%). Hence, our first keyphrase property concerns the size of keyphrases.

Property 1 *Keyphrases bear the minimum information representing an important topic or concept (e.g. "T-2 Buckeye" instead of "two-seat T-2 Buckeye").*

Second, we observe from both English and French datasets that almost every keyphrase contains one or more nouns and half of the keyphrases are modified using one or more adjectives. Among these adjectives, it is important to note the usage of relational adjectives. Although they are less used than non-relational adjectives, their similar properties to nouns (Bally, 1944) make them more likely to be relevant keyphrase modifiers than other adjectives. Indeed, adjectives, such as "huge", are not relevant keyphrase modifiers, but some candidate selection methods may, for example, only select "huge wildfires", leading to a miss of the rightful candidate "wildfires". In opposition, relational adjectives, such as "presidential", are noun substitutes that have a classificatory or taxonomic meaning (McNally and Boleda, 2004), which makes them more relevant as keyphrase modifiers.

Property 2 *Keyphrases are mostly nouns (e.g. "storms") that can be modified by one or more adjectives (e.g. "<u>annual</u> hurricane forecast").*

[1] We observed that keyphrases containing one word are mostly nouns or proper nouns. Hence, we only show the POS tag statistics of the multi-word keyphrases.

Statistic	DUC	SemEval	DEFT
Documents			
Number	208	144	141
Tokens/document	912.0	5,134.6	7,276.7
Keyphrases/document	8.1	15.4	5.4
Missing keyphrases	3.9%	13.5%	18.2%
Keyphrases			
Unigrams	17.1%	20.2%	60.2%
Bigrams	60.8%	53.4%	24.5%
Trigrams	17.8%	21.3%	8.8%
N-grams (N $\geq$ 4)	4.3%	5.2%	6.6%
Multi-word keyphrases			
Containing noun(s)	94.5%	98.7%	93.3%
Containing proper noun(s)	17.1%	4.3%	6.9%
Containing (non-relational) adjective(s)	32.5%	40.5%	29.9%
Containing relational adjective(s)	20.0%	11.1%	37.2%
Containing verb(s)	1.0%	4.0%	1.0%
Containing adverb(s)	1.6%	0.7%	1.3%
Containing preposition(s)	0.3%	1.5%	31.2%
Containing determiner(s)	0.0%	0.0%	20.4%

Table 1: Statistics of the training datasets. Missing keyphrases are keyphrases that do not occur in the documents.

To give an idea of the observed POS tag patterns, Table 2 shows the five most frequent POS tag patterns for English and French keyphrases.

	Pattern	Example
English	`Nc Nc`	*"hurricane expert"*
	`nA Nc`	*"turbulent summer"*
	`Nc`	*"storms"*
	`rA Nc`	*"Chinese earthquake"*
	`nA Nc Nc`	*"annual hurricane forecast"*
French	`Nc`	*"patrimoine" ("cultural heritage")*
	`Np`	*"Indonésie" ("Indonesia")*
	`Nc rA`	*"tradition orale" ("oral tradition")*
	`Nc nA`	*"anthropologie réflexive" ("reflexive anthropology")*
	`Nc Sp D Nc`	*"conservation de la nature" ("nature conservation")*
	`Nc Sp Nc`	*"traduction en anglais" ("English translation")*

Table 2: Frequent POS tag patterns. POS tags belong to the Multex format, except `rA` and `nA` which stands for, respectively, *relational adjective* and *non-relational adjective*.

3 Candidate Selection

In this section, we present the textual units that are commonly used as keyphrase candidates and discuss their consistency regarding the properties inferred in Section 2. We also present a new method to selected refined noun phrases as keyphrase candidates.

N-grams are ordered sequences of n words, where n is usually set to 1 up to 3 (Witten et al., 1999). Extracting n-grams has the benefit to provide almost every candidates that actually match reference keyphrases (maximum recall), but the counterpart is that it also provides a huge amount of unrelevant candidates. Therefore, Witten et al. (1999) propose to select only n-grams that do not contain a stop word (conjunction, preposition, determiner or common word) at their beginning or end. Filtered n-gram candidates are grammatically uncontrolled and do not fit properties 1 and 2.

Textual units matching given POS tag patterns are textual units of specific syntactic forms. Extracting such textual units ensures grammaticality and precisely defines the nature of the candidates. In previous work, Hulth (2003) experiments with the most frequent POS tag patterns of her training data[2], whereas other researchers select the longest sequences of nouns, proper nouns and adjectives, namely the longest NPs (Hasan and Ng, 2010). Candidates selected using both approaches fit both properties 1 and 2. However, the first approach needs training data and is, therefore, not suitable to every situation.

NP-chunks are non-recursive noun phrases. Hulth (2003) uses them in her work and argues that they are less arbitrary and more linguistically justified than other candidates such as n-grams. Also, as NP-chunks are non-recursive (hence minimal) noun phrases, they are consistent with both properties 1 and 2.

As a contribution to the candidate selection, we propose to extract **refined noun phrases** (refined NPs) by adding a decision process during the selection of noun, proper noun and adjective sequences. According to the fact that some adjectives do not add important information to the noun phrase they modify (e.g. "huge wildfires"), we propose to select noun phrases and to keep their adjective modifier (if they have one) under specific conditions:

1. the adjective is a relational adjective, or
2. the adjective co-occurs at least two times with the noun phrase it modifies.

This method can be seen as a refinement of the selection of the longest noun phrases and NP-chunks. It also fits both properties 1 and 2.

4 Keyphrase Extraction

Once candidates are select, the second step of the keyphrase extraction task is to classify them, as "keyphrase" or "non-keyphrase", or rank them in order to extract the k bests as keyphrases. In this section, we detail the three keyphrase extraction methods that we use in our study. Two are different unsupervised methods (ranking methods) and one is a supervised method (classification method).

TF-IDF (Spärck Jones, 1972) is a weighting scheme that represents the significance of a word in a given document. Significant words must be both frequent in the document and specific to it. The specificity of a word is determined based on a collection of documents. That the lower is the amount of documents containing a given word, the higher is its specificity. Keyphrase candidates are scored according to the sum of the TF-IDF weights of their words and the k best candidates are extracted as keyphrases.

TopicRank (Bougouin et al., 2013) aims to extract keyphrases that best represent the main topics of a document. Keyphrase candidates are clustered into topics using a stem overlap similarity, each topic is scored using the TextRank random walk algorithm (Mihalcea and Tarau, 2004) and one representative keyphrase is extracted from each of the k best ranked topics.

KEA (Witten et al., 1999) is a supervised method that uses a Naive Bayes classifier to extract keyphrases. The classifier combines two feature probabilities to predict whether a candidate is a "keyphrase" or a "non-keyphrase". The two features are the TF-IDF weight[3] of the candidate and the position of its first appearance in the document.

[2]Frequent patterns are the ones that appear at least ten times in the training data.
[3]The TF-IDF weight computed for KEA is based on candidate frequency, not word frequency.

5 Experiments

To observe the impact of the candidate selection methods, we perform two experiments. First, we compare the quality of the candidate selection methods with the set of reference keyphrases. Second, we compare their impact on the keyphrase extraction task, applying them to TF-IDF, TopicRank and KEA.

5.1 Evaluation Measures

To quantify the capacity of the keyphrase candidate selection methods to provide suitable candidates and avoid irrelevant ones, we compute the number of selected candidates (Cand./Doc.) and confront it with the maximum recall (Rmax) that can be achieved. To do so, we compute a quality ratio (QR):

$$\text{QR} = \frac{\text{Rmax}}{\text{Cand./Doc.}} \times 100 \qquad (1)$$

The higher is the QR value of a candidate set, the better is its quality.

To evaluate the performance of the keyphrase extraction methods, we use the common precision (P), recall (R) and f-score (F) measures, when a maximum of 10 keyphrases are extracted.

5.2 Preprocessing

For each dataset, we apply the following preprocessing steps: sentence segmentation, word tokenization and Part-of-Speech tagging. For sentence segmentation, we use the PunktSentenceTokenizer provided by the python Natural Language ToolKit (Bird et al., 2009). For word tokenization, we use the NLTK TreebankWordTokenizer for English and the Bonsai word tokenizer[4] for French. As for Part-of-Speech tagging, we use the Stanford POS tagger (Toutanova et al., 2003) for English and MElt (Denis and Sagot, 2009) for French.

5.3 Candidate Selection

This section presents an intrinsic evaluation of the candidate selection methods described in Section 3. The aim is to compare the methods in terms of quantity of selected candidates and percentage of reference keyphrases that can be found in the best case (maximum recall).

5.3.1 Method Settings

For each candidate selection method presented in section 3, we configure one method with the parameters that best fit properties 1 and 2.

According to Property 1, we test a **filtered n-gram selection** method that provides small size n-grams: $n = \{1..3\}$. The stop words used for the filtering are part of the IR Multilingual Resources[5] provided by the University of Neuchâtel (UniNE).

Following both Property 2 and previous work (Hasan and Ng, 2010), we use **pattern matching** to select the longest noun phrases (longest NPs), i.e. the longest sequences of nouns, proper nouns and adjectives.

The **NP-chunk selection** is also performed using pattern matching. Only basic patterns are used:

- `Np+ | (A+ Nc) | Nc+`, for English datasets;
- `Np+ | (A? Nc A+) | (A Nc) | Nc+`, for French datasets.

[4] The Bonsai word tokenizer is a tool provided with the Bonsai PCFG-LA parser: `http://alpage.inria.fr/statgram/frdep/fr_stat_dep_parsing.html`.

[5] `http://members.unine.ch/jacques.savoy/clef/index.html`

Method	DUC			SemEval			DEFT		
	Cand./Doc.	Rmax	QR	Cand./Doc.	Rmax	QR	Cand./Doc.	Rmax	QR
{1..3}-grams	596.2	90.8	15.2	2,580.5	72.2	2.8	4,070.2	74.1	1.8
Longest NPs	155.6	88.7	57.0	646.5	62.4	9.7	914.5	61.1	6.7
NP-chunks	149.9	76.0	50.7	598.4	56.6	9.5	812.3	63.0	7.8
Refined NPs	143.1	85.2	**59.5**	563.4	60.7	**10.8**	670.0	58.6	**8.7**

Table 3: Candidate selection statistics.

Method	DUC			SemEval			DEFT		
	P	R	F	P	R	F	P	R	F
{1..3}-grams	14.3	19.0	16.1	9.0	6.0	7.2	6.7	12.5	8.6
Longest NPs	**24.2**	**31.7**	**27.1**	11.7	7.9	9.3	9.5	17.6	12.1
NP-chunks	21.1	28.1	23.8	11.9	8.0	9.5	9.6	17.9	12.3
Refined NPs	24.2	31.6	27.0	**12.3**	**8.4**	**9.8**	**10.0**	**18.3**	**12.7**

Table 4: Comparison of candidate selection methods, when 10 keyphrases are extracted by **TF-IDF**.

The **refined NPs** are also selected using pattern matching. The patterns we use are related to the position of relational adjectives in the target language:

- `A? (Nc | Np)+`, for English datasets;
- `(Nc | Np)+ A?`, for French datasets.

Relational adjectives are detected using the WordNet lexical database (Miller, 1995) for English and its French translation, WoNeF (Pradet et al., 2013), for French. Also, to detect other (potential) relational adjectives, we use a list of two-size suffixes automatically built from WordNet and WoNeF relational adjectives.

5.3.2 Result Analysis

Table 3 shows the results of the candidate selection methods. The selection of n-grams provides a huge amount of candidates and allows a near perfect maximum recall[6], whereas the other candidate selection methods provide less candidates and allow a lower maximum recall. However, for the selection of longest NPs, NP-chunks and refined NPs, the maximum recall does not significantly decrease compared to the number of selected candidates and according to the quality ratio, the method that selects better candidates is the one selecting refined NPs, followed by the ones selecting longest NPs and NP-chunks.

[TODO Add an analysis of the selected candidates (as in section 2)?]

5.4 Keyphrase Extraction

This section presents an extrinsic evaluation of the candidate selection methods. The aim is to observe the impact of the candidate selection methods on the keyphrase extraction task.

5.4.1 Result Analysis

Tables 4, 5 and 6 show the performance of respectively TF-IDF, TopicRank and KEA when they extract keyphrases from keyphrase candidates provided by each candidate selection method.

Gloabally, our method, followed by the selection of longest NPs and NP-chunks, is the one that induces the best performance for each method. This confirms that small candidate sets of high quality are better than exhaustive candidate sets such as n-grams. However, the results of KEA are more stable than the

[6] According to the amount of missing keyphrases of the test sets, the maximum recall that can be achieved is 97.2% for DUC, 87.9% for SemEval and 88.9% for DEFT.

Method	DUC			SemEval			DEFT		
	P	R	F	P	R	F	P	R	F
{1..3}-grams	7.8	10.7	8.9	9.5	6.7	7.7	6.2	11.4	8.0
Longest NPs	17.7	23.2	19.8	11.6	7.9	9.3	**11.6**	**21.5**	**14.9**
NP-chunks	13.3	21.5	18.3	11.7	8.0	9.4	11.1	20.7	14.4
Refined NPs	**17.8**	**23.5**	**20.0**	**12.1**	**8.3**	**9.8**	11.4	21.0	14.6

Table 5: Comparison of candidate selection methods, when 10 keyphrases are extracted by **TopicRank**.

Method	DUC			SemEval			DEFT		
	P	R	F	P	R	F	P	R	F
{1..3}-grams	12.0	16.6	13.7	19.4	13.7	15.9	13.4	25.3	17.3
Longest NPs	14.5	19.9	16.5	19.6	13.7	16.0	14.1	26.3	18.1
NP-chunks	13.5	18.6	15.4	19.5	13.7	16.0	14.3	26.8	18.4
Refined NPs	**14.5**	**20.1**	**16.6**	**20.5**	**14.4**	**16.8**	**14.4**	**27.1**	**18.6**

Table 6: Comparison of candidate selection methods, when 10 keyphrases are extracted by **KEA**.

results of TF-IDF and TopicRank. KEA's learning step makes it less sensitive to irrelevant candidates that produce noise affecting the unsupervised methods. [TODO Too short.]

6 Conclusion

In this paper, we argued that the candidate selection is a critical step of the keyphrase extraction task and studied the impact of various candidate selections over different keyphrase extraction methods.

According to the reference keyphrases of three standard datasets, we inferred two general keyphrase properties: (1) keyphrases are mostly noun phrases, which (2) are short (one, two or three words) and bear only sufficient information. Among the candidate selection methods, selecting textual units matching predefined patterns is the best ways to obtain keyphrase candidates that fit both properties.

To confirm our assertions, we conducted two experiments to compare the quality of the candidate sets and observe their impact on the performance of different keyphrase extraction methods. Experimental results show that unsupervised methods do not have stable results depending on the candidate selection methods. We found that concise candidate sets allowing a high maximum recall induce better results than huge candidate sets allowing a highest maximum recall. In other words, the quality of a candidate set prevails over its exhaustivity.

We also presented a new candidate selection method that defines noun phrase patterns and filter their adjective modifier in order to keep only relevant ones. Based on the assumption that relational adjectives act as nouns, we decided to keep them, along with the adjectives that frequently modifies the same noun phrase.

Our work states that adjectives must not necessary be keyphrase modifiers and we showed that simple linguistic filters can increase the quality of the selected candidates. Hence, future work on candidate selection might focus on more complex linguistic and/or statistical filtering methods for adjectives.

Acknowledgements

The authors would like to thank the anonymous reviewers for their useful advice and comments. This work was supported by the French National Research Agency (TermITH project – ANR-12-CORD-0029).

References

Charles Bally. 1944. Linguistique générale et linguistique française. *Paris: Ernest Leroux*.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural Language Processing with Python*. O'Reilly Media.

Adrien Bougouin, Florian Boudin, and Béatrice Daille. 2013. Topicrank: Graph-Based Topic Ranking for Keyphrase Extraction. In *Proceedings of the 6th International Joint Conference on Natural Language Processing (IJCNLP)*, pages 543–551, Nagoya, Japan, October. Asian Federation of Natural Language Processing.

Pascal Denis and Benoît Sagot. 2009. Coupling an Annotated Corpus and a Morphosyntactic Lexicon for State-of-the-Art POS Tagging with Less Human Effort. In *Proceedings of the 23rd Pacific Asia Conference on Language, Information and Computation (PACLIC)*, pages 110–119, Hong Kong, December. City University of Hong Kong.

Ernesto D'Avanzo and Bernado Magnini. 2005. A Keyphrase-Based Approach to Summarization: the LAKE System at DUC-2005. In *Proceedings of DUC 2005 Document Understanding Conference*.

Kazi Saidul Hasan and Vincent Ng. 2010. Conundrums in Unsupervised Keyphrase Extraction: Making Sense of the State-of-the-Art. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 365–373, Stroudsburg, PA, USA. Association for Computational Linguistics.

Anette Hulth. 2003. Improved Automatic Keyword Extraction Given More Linguistic Knowledge. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, pages 216–223, Stroudsburg, PA, USA. Association for Computational Linguistics.

Su Nan Kim, Min-Yen. Kan, and Timothy Baldwin. 2009. An Unsupervised Approach to Domain-Specific Term Extraction. In *Proceedings of the 2009 Australasian Language Technology Association Workshop*.

Su Nam Kim, Olena Medelyan, Min-Yen Kan, and Timothy Baldwin. 2010. SemEval-2010 task 5: Automatic Keyphrase Extraction from Scientific Articles. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 21–26, Stroudsburg, PA, USA. Association for Computational Linguistics.

Louise McNally and Gemma Boleda. 2004. Relational Adjectives as Properties of Kinds. *Empirical Issues in Formal Syntax and Semantics*, 5:179–196.

Olena Medelyan and Ian H. Witten. 2008. Domain-Independent Automatic Keyphrase Indexing with Small Training Sets. *Journal of the American Society for Information Science and Technology*, 59(7):1026–1040, may.

Rada Mihalcea and Paul Tarau. 2004. TextRank: Bringing Order Into Texts. In Dekang Lin and Dekai Wu, editors, *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 404–411, Barcelona, Spain, July. Association for Computational Linguistics.

George A. Miller. 1995. WordNet: a Lexical Database for English. *Communications of the Association for Computational Linguistics*, 38(11):39–41.

Paul Over. 2001. Introduction to DUC-2001: an Intrinsic Evaluation of Generic News Text Summarization Systems. In *Proceedings of DUC 2001 Document Understanding Conference*.

Patrick Paroubek, Pierre Zweigenbaum, Dominic Forest, and Cyril Grouin. 2012. Indexation libre et contrôlée d'articles scientifiques. Présentation et résultats du défi fouille de textes DEFT2012 (Controlled and Free Indexing of Scientific Papers. Presentation and Results of the DEFT2012 Text-Mining Challenge) [in French]. In *Proceedings of the Joint Conference JEP-TALN-RECITAL 2012, Workshop DEFT 2012: DÉfi Fouille de Textes (DEFT 2012 Workshop: Text Mining Challenge)*, pages 1–13, Grenoble, France, June. ATALA/AFCP.

Quentin Pradet, Jeanne Baguenier Desormeaux, Gaël de Chalendar, and Laurence Danlos. 2013. WoNeF : amélioration, extension et évaluation d'une traduction française automatique de WordNet. In *Actes de la 20e conférence sur le Traitement Automatique des Langues Naturelles (TALN'2013)*, pages 76–89, Les Sables d'Olonne, France.

Karen Spärck Jones. 1972. A Statistical Interpretation of Term Specificity and its Application in Retrieval. *Journal of Documentation*, 28(1):11–21.

Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. 2003. Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, pages 173–180, Stroudsburg, PA, USA. Association for Computational Linguistics.

Peter D. Turney. 2000. Learning Algorithms for Keyphrase Extraction. *Information Retrieval*, 2(4):303–336, may.

Xiaojun Wan and Jianguo Xiao. 2008. Single Document Keyphrase Extraction Using Neighborhood Knowledge. In *Proceedings of the 23rd National Conference on Artificial Intelligence - Volume 2*, pages 855–860. AAAI Press.

Ian H. Witten, Gordon W. Paynter, Eibe Frank, Carl Gutwin, and Craig G. Nevill Manning. 1999. KEA: Practical Automatic Keyphrase Extraction. In *Proceedings of the 4th ACM Conference on Digital Libraries*, pages 254–255, New York, NY, USA. ACM.