Extracting Keyphrases from Research Papers using Citation Networks

Sujatha Das Gollapalli and Cornelia Caragea
Computer Science and Engineering
University of North Texas
Email: gsdas@cse.psu.edu, ccaragea@unt.edu

Abstract

Keyphrases for a document concisely describe the document using a small set of phrases. Keyphrases were previously shown to improve several document processing and retrieval tasks. In this work, we study keyphrase extraction from research papers by leveraging citation networks. We propose CiteTextRank for keyphrase extraction from research articles, a graph-based algorithm that incorporates evidence from both a document's content as well as the *contexts* in which the document is referenced within a citation network. Our model obtains significant improvements over the state-of-the-art models for this task. Specifically, on several datasets of research papers, CiteTextRank improves precision at rank 1 by as much as 9-20% over state-of-the-art baselines.

Introduction

The important parts or "concepts" in documents often are not provided with the documents. Instead, these concepts need to be gleaned from the many details in documents. Keyphrase extraction is defined as the problem of automatically extracting descriptive phrases or concepts from a document. **Keyphrases** act as a concise summary of a document and have been successfully used in several data mining applications such as query formulation, document clustering, recommendation, and summarization (Jones and Staveley 1999; Zha 2002; Hammouda, Matute, and Kamel 2005; Pudota et al. 2010). In this paper, we address the problem of keyphrase extraction from research papers.

Research papers (also referred to as documents or papers in this paper) enable the sharing and dissemination of scientific discoveries. During these "big data" times, keyphrases associated with research papers can allow for *efficient* processing of more information in less time for top-level data mining applications on research document collections, e.g., topic tracking, information filtering, and search. As a result, several online digital libraries such as the ACM digital library have started to impose the requirement for author-input keyphrases (for example, via free-form tags or selection from a pre-defined list of keyphrases) for documents acquired by them. However, keyphrases have not been integrated into all established mechanisms of data sharing and organization. For example, the official AAAI website (http://www.aaai.org/) does not provide keyphrases associated with research papers published in the AAAI conference. For these scenarios, automatic techniques are required for extracting keyphrases from research documents.

Most existing keyphrase extraction techniques used only the textual content of the target document (Mihalcea and Tarau 2004; Liu et al. 2010). Recently, Wan and Xiao (2008) proposed a model that incorporates a local neighborhood of a document for extracting keyphrases. However, their neighborhood is limited to textually-similar documents, where the cosine similarity between the *tf-idf* vectors of documents is used to compute documents' similarity. We posit that, in addition to a document's textual content and textually-similar neighbors, other informative neighborhoods exist in research document collections that have the potential to improve keyphrase extraction.

Scientific research papers typically propose new problems or extend the state-of-the-art for existing research problems. Consider a research paper d. It is common to find in d, relevant, previously-published research papers cited in appropriate *contexts*. Such citations between research papers give rise to a large network of interlinked documents, commonly referred to as the *citation network*.

In a citation network, information flows from one paper to another via the citation relation (Shi, Leskovec, and McFarland 2010). This information flow as well as the influence of one paper on another are specifically captured by means of *citation contexts* (i.e., short text segments surrounding a paper's mention). These contexts are not arbitrary, but they serve as "micro summaries" of a cited paper. Figure 1 shows an anecdotal example illustrating this behavior using the 2010 best paper award winner in the World Wide Web conference (Paper 1) and its citation network neighbor (Paper 2). Notice in this example, the large overlap in the author-specified keywords[1] and the citation contexts in Figure 1.

Can citation networks improve the performance of keyphrase extraction? Since citation contexts capture how a paper influences another in the network along various aspects such as topicality, domain of study, and algorithms, how can we use these "micro summaries" in keyphrase ex-

Copyright © 2014, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

[1]We use the term "keywords" and "keyphrases" interchangeably and the terms can refer to both single words or phrases.

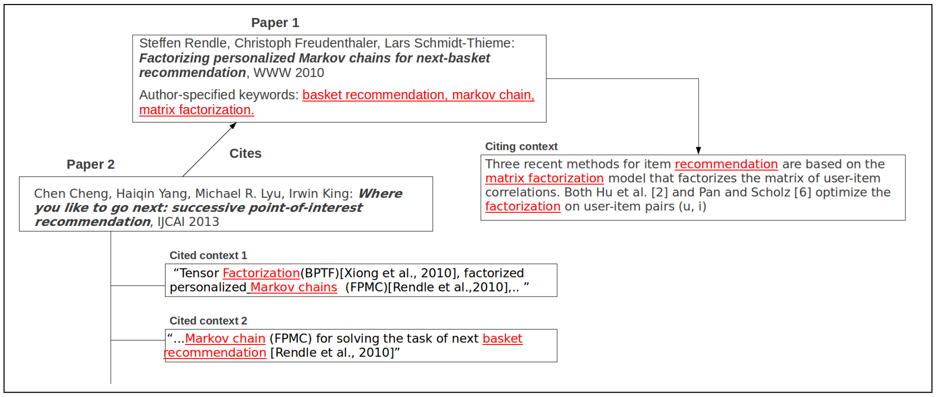

Figure 1: A small citation network.

traction models? We specifically address these questions with our research agenda in this paper.

Our contributions are as follows:

1. We propose a novel approach to keyphrase extraction from scientific documents using the citation network information. In particular, we effectively leverage the contexts in which a document is referenced in citation networks for extracting keyphrases.
2. We design CiteTextRank, a fully unsupervised graph-based algorithm that incorporates evidence from multiple sources (citation contexts as well as document content) in a flexible manner to score keywords that are later used to score keyphrases.
3. We experimentally validate CiteTextRank on several representative datasets and show statistically significant improvements over existing state-of-the-art models for keyphrase extraction.
4. Additionally, as part of our contributions, we created three representative datasets that will be made available to further research in the keyphrase extraction community.

Our research hypothesis and motivation were just presented in this section. In the following section, we briefly summarize related, state-of-the-art approaches to keyphrase extraction. CiteTextRank is described in detail in the "Proposed Model" section. Finally, we present our datasets, experiments and results before concluding the paper.

Related work

Keyphrase extraction was previously studied using both supervised and unsupervised techniques for different types of documents including scientific abstracts, newswire documents, meeting transcripts, and webpages (Frank et al. 1999; Hulth 2003; Nguyen and Kan 2007; Liu et al. 2009; Marujo et al. 2013). The recent SemEval 2010 Shared Task was focused on comparing keyphrase extraction systems for scientific articles (Kim et al. 2010; 2013), indicating once again the significance of this problem.

Supervised techniques use annotated documents with "correct" keyphrases to train classifiers for discriminating keyphrases extracted from a document (Frank et al. 1999; Turney 2000; Hulth 2003). In unsupervised keyphrase extraction, domain-specific knowledge and various measures such as term frequencies, inverse document frequencies, topic proportions, etc. are used to score terms in a document that are later aggregated to obtain scores for phrases (Mihalcea and Tarau 2004; Liu et al. 2010; Boudin 2013).

The PageRank algorithm is widely-used in keyphrase extraction models. Other centrality measures such as betweenness and degree centrality were also previously studied for keyphrase extraction (Palshikar 2007). However, based on recent experiments in (Kim et al. 2013; Kim and Kan 2009) and (Boudin 2013), the PageRank family of methods and *tf-idf* based scoring can be considered the state-of-the-art for unsupervised keyphrase extraction.

Mihalcea and Tarau (2004), were the first to propose TextRank, for scoring keyphrases using the PageRank values obtained on a word graph built from the adjacent words in a document. Wan and Xiao (2008) extended the TextRank approach to SingleRank by adding edges between words within a window size greater than 2 and edge weights in the graph based on co-occurrence between words. Unlike the TextRank and SingleRank models, where only the content of the target document are used for keyphrase extraction, textually-similar neighboring documents are included in the scoring process in ExpandRank (Wan and Xiao 2008).

In contrast to the approaches above, we present a model for keyphrase extraction from research papers that are embedded in citation networks. The underlying algorithm of our model is PageRank applied to word graphs constructed from target papers and their local neighborhood in a citation network. In addition, unlike the approaches so far, our model incorporates multiple neighborhoods and includes a flexible way to incorporate different weights for each neighborhood.

Recently, *social tagging* has been prevalent in Web 2.0 applications. Similar to keyphrase extraction, tag recommendation systems are designed to predict descriptive terms or *tags* for organizing and sharing Web resources. However, in contrast with our problem setup, most of these systems are designed for web objects that also have non-textual content. In addition, previous behavior of users in interactions with these objects can be used for recommending tags (Song, Zhang, and Giles 2008; Rendle et al. 2009; Yin et al. 2010). Due to these differences, we do not discuss tag recommendation further.

Proposed Model

Preliminaries

Definition 1 (Keyphrase Extraction) Given a target document d, the objective of the keyphrase extraction task is to extract a ranked list of candidate words or phrases from d that best represent d.

Algorithms for unsupervised keyphrase extraction commonly involve three steps (Hasan and Ng 2010):

1. Candidate words or lexical units are extracted from the textual content of the target document by applying stopword and parts-of-speech filters. Only noun and adjectives that are likely to be keyphrases are retained in this step.
2. Next, candidate words are scored based on some criterion. For example, in the TFIDF scoring scheme, a candidate word score is the product of its frequency in the document and its inverse document frequency in the collection.
3. Finally, consecutive words, phrases or n-grams are scored by using the sum of scores of individual words that comprise the phrase (Wan and Xiao 2008). The top-scoring phrases are output as predictions (the keyphrases for the document).

In what follows, we describe CiteTextRank (CTR), our fully unsupervised, graph-based model that explicitly incorporates information from *citation contexts* while scoring candidate words in step (2).

CiteTextRank

Let d be the target document for keyphrase extraction and $\mathcal{C}$ be a citation network such that $d \in \mathcal{C}$.

Definition 2 (Contexts) A *cited context* for document d is defined as a context in which d *is cited* by some paper d_i in the network. A *citing context* for d is defined as a context in which d *is citing* some paper d_j in the network. The content of d comprises its *global context*.

If a document d_1 cites a document d_2 in multiple contexts, then we aggregate all such contexts and simply refer to them as the citing or cited context (as applicable).

Let T represent the types of available contexts for d. These types include the *global* context of d, ${\mathcal{N}_d}^{Ctd}$, the set of *cited* contexts for d, ${\mathcal{N}_d}^{Ctg}$, the set of *citing* contexts for d, and textually-similar global contexts, ${\mathcal{N}_d}^{Sim}$. We construct an undirected graph, $G = (V, E)$ for d as follows:

1. For each unique candidate word extracted from all available contexts of d, add a vertex in G.
2. Add an undirected edge between two vertices v_i and v_j if the words corresponding to these vertices occur within a window of w contiguous tokens in any of the contexts.
3. The weight w_{ij} of an edge $(v_i, v_j) \in E$ is given as

$$w_{ij} = w_{ji} = \sum_{t \in T} \sum_{c \in C_t} \lambda_t \cdot \text{cossim}(c, d) \cdot \#_c(v_i, v_j) \quad (1)$$

where $\text{cossim}(c, d)$ is the cosine similarity between the *tf-idf* vectors of any context c of d and d (Manning, Raghavan, and Schütze 2008); $\#_c(v_i, v_j)$ is the co-occurrence frequency of words corresponding to v_i and v_j in context c; C_t is the set of contexts of type $t \in T$; and λ_t is the weight for contexts of type t.

Finally, we score the vertices in G (and the corresponding candidate words) using the PageRank algorithm (Page et al. 1999). That is, the score s for vertex v_i is obtained by recursively computing the equation:

$$s(v_i) = (1 - \alpha) + \alpha \sum_{v_j \in Adj(v_i)} \frac{w_{ji}}{\sum_{v_k \in Adj(v_j)} w_{jk}} s(v_j) \quad (2)$$

where α is the damping factor typically set to 0.85 (Haveliwala et al. 2003).

Unlike simple graph edges with fixed weights, notice that our equations correspond to *parameterized* edge weights. We incorporate the notion of "importance" of contexts of a certain type using the λ_t parameters. For instance, one might assign higher importance to citation contexts over global contexts, or cited over citing contexts. One way to visualize our edges is to imagine the two vertices in the underlying graph to be connected using multiple edges of different types. For example, in Figure 2, the two edges between "logic" and "programming" could correspond to *cited* and *global* contexts respectively.

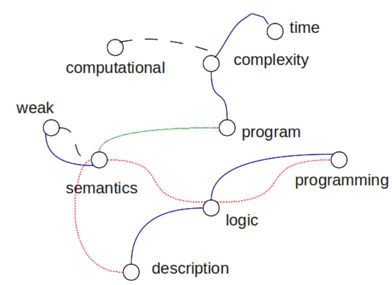

Figure 2: A small word graph. We show the edges added due to different context types using different colors/line-styles

Intuitively, a context that is more similar to the target document should be assigned higher weight during keyphrase extraction. In addition, we would like to score edges between word pairs that co-occur frequently higher than those that co-occur rarely. These two aspects are captured by the cosine similarity and co-occurrence components in the edge weights of Equation (1).

The vertices in G can be scored using any graph centrality-based measures. However, PageRank models were shown to be state-of-the-art in previous research involving word graphs for keyphrase extraction (Mihalcea and Tarau 2004; Liu et al. 2010; Boudin 2013). In particular, the PageRank score for a node (or vertex) provides a measure of importance for the node in the graph by taking into account global information computed recursively from the entire graph (Brin and Page 1998; Haveliwala et al. 2003).

PageRank uses the notion of "voting" between vertices to assign "importance" scores for vertices (Page et al. 1999). If a node links to another one, it is casting a vote to that other node. Nodes recursively cast votes across their links, where the votes are weighted by the current estimate of their nodes' PageRank. Hence, the score associated with a vertex depends on the number of votes that are cast for it, as well as the importance of nodes casting these votes (Brin and Page 1998). For graphs where vertices correspond to words,

Conference	#Titles(Org)	#Titles(CiteSeer)	#Queries	AvgKeywords	AvgCitingContexts	AvgCitedContexts
AAAI	5676	2424	93	4.15	9.77	13.95
UMD	490	439	163	3.93	20.15	34.65
WWW	2936	1350	406	4.81	15.91	17.39
KDD	1829	834	335	4.09	18.85	16.82

Table 1: Summary of datasets, #Queries represent the number of documents for which both citing, cited contexts were extracted from CiteSeer and for which the "correct" keyphrases are available. All datasets are available upon request.

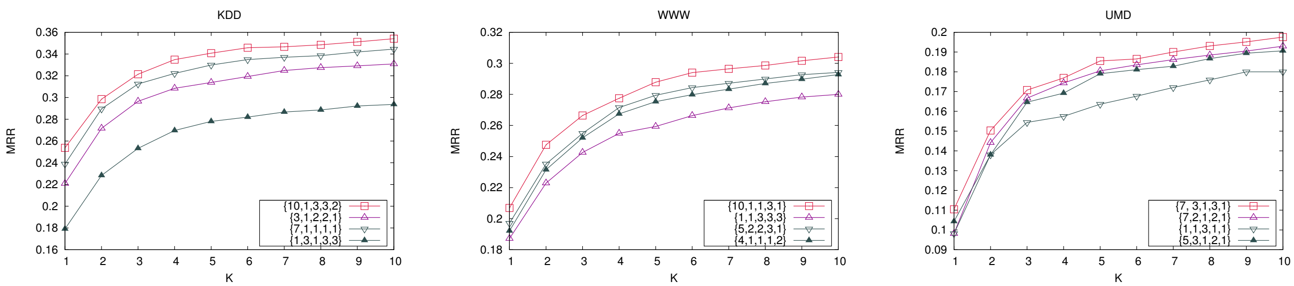

Figure 3: Parameter tuning for CiteTextRank. Sample configurations are shown. Setting {a,b,c,d,e} indicates that the window parameter is set to 'a' with 'b', 'c', 'd', 'e' as weights for textually-similar neighbors, cited, citing, and global contexts of a document, respectively.

PageRank scores therefore capture the intuition that *a word is important if it is linked to many words, some of which are important as well*.

In CiteTextRank, unlike other PageRank-based keyphrase extraction models (Mihalcea and Tarau 2004; Wan and Xiao 2008), we build the graph for computing scores using information from multiple contexts. Moreover, the λ_t parameters provide means to tune the importance of each type of context. We now experimentally show that CTR effectively captures the notion of word importance across multiple global and citation contexts.

Experiments

Datasets and evaluation measures

Existing benchmark datasets for keyword extraction do not contain citation network information (Hulth 2003; Hasan and Ng 2010; Kim et al. 2010). In order to test the performance of CiteTextRank, we need gold-standard annotated datasets of research papers and their associated citation networks. We constructed three such datasets. The first two are proceedings of the last ten years of: (1) the ACM Conference on Knowledge Discovery and Data Mining (KDD), and (2) the World Wide Web Conference (WWW). The third dataset (referred to as UMD in this paper) was made available by Lise Getoor's research group at the University of Maryland[2]. This dataset was previously used to study document classification (Lu and Getoor 2003) and citation recommendation (Kataria, Mitra, and Bhatia 2010).

For all datasets, we used paper titles to query the CiteSeerX[3] digital library (Giles, Bollacker, and Lawrence 1998) and retrieve their corresponding pdf documents, as well as the *cited* and *citing* contexts for each document. We collected the titles of all papers published in KDD and WWW directly from these conferences' websites. From all pdf documents found in CiteSeerX, we retained in our datasets only those for which we were able to extract the abstract and the author-input keywords (from the **Keywords** field). These keywords represent the *gold-standard* for evaluation.

In addition, we evaluated CTR for keyword extraction from papers published in the previous years of AAAI[4]. We adopted the same procedure that was used to create our other datasets to obtain the citation network for AAAI papers. However, since author-specified keywords are not available with AAAI papers, we manually examined and annotated about 100 randomly selected AAAI papers with keywords.

Our datasets are summarized in Table 1. This table shows the original number of titles in each collection along with the number of titles we were able to map to CiteSeerX documents using title search. The number of documents having cited and citing contexts and abstracts as well as the average numbers of keywords, citing, and cited contexts are listed in this table. On average, authors describe a given research paper with a small set of 3-5 keywords making it a challenging task to rank the appropriate keywords in the top positions.

Context lengths. We used citation contexts of papers as obtained from CiteSeerX directly. CiteSeerX provides a context of 50 words on each side of a citation mention in a paper. According to a previous study on the impact of various citation context length on information retrieval tasks by Ritchie (2008), a fixed window length of about 100 words was found to be effective in general. Accurate extraction and categorization of citation contexts is a subject of current research (Teufel 1999; Abu-Jbara and Radev 2012) and we plan to study the effect of context lengths on keyphrase extraction in future. For the *global context* of a document, we use its title and abstract. The choice of not considering the

[2]http://www.cs.umd.edu/∼sen/lbc-proj/LBC.html

[3]http://citeseerx.ist.psu.edu

[4]http://www.aaai.org/Library/AAAI/aaai-library.php

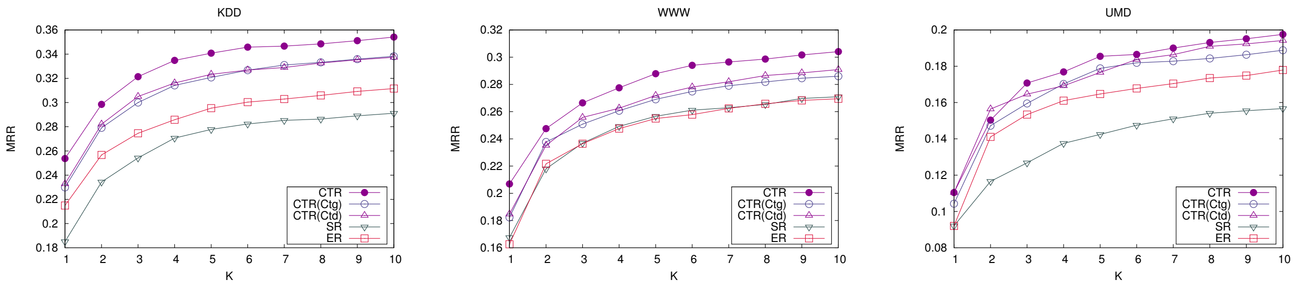

Figure 4: Effect of citation network information on keyphrase extraction. CiteTextRank (CTR) that uses citation network neighbors is compared with ExpandRank (ER) that uses textually-similar neighbors and SingleRank (SR) that only uses the target document content.

entire text is due to the observation that scientific papers contain details such as discussion of results, experimental design, and notation that do not provide additional benefits for extracting keyphrases. Therefore, similar to previous works, we do not use the entire text (Mihalcea and Tarau 2004; Hulth 2003).

Evaluation measures. As in previous works, we use the measures: precision, recall, F1, and mean reciprocal rank (MRR) for evaluating our model (Manning, Raghavan, and Schütze 2008). In particular, we use MRR curves to illustrate our experimental findings. MRR is a standard measure used in ranked retrieval tasks that gives the averaged ranking of the first correct prediction. That is, $MRR = \frac{1}{|Q|} \sum_{q=1...|Q|} \frac{1}{r_q}$ where r_q refers to the rank at which the first correct prediction was found for query $q \in Q$ (here Q is the set of target documents). For computing "@k" numbers (such as MRR@k), we only examine the top-k predictions. We use *average* k to refer to the average number of keywords for a particular dataset as listed in Table 1. For example, *average* $k = 5$ for the WWW dataset.

Results and Observations

Our experiments are organized around several questions as detailed below.

How sensitive is CiteTextRank to its parameters? CTR has two sets of parameters, the window size w that determines how the edges are added between candidate word nodes in the graph and the λ_t values that determine the weight of each context type. To illustrate the effect of parameters in the CiteTextRank model, we randomly chose a few top-performing and not-so-well performing parameter configurations (w and λ_t values) from our parameter sweep experiments and plot them in Figure 3. Changing the λ_t values corresponds to assigning different weights to the global context, neighbors based on content similarity, and citing and cited contexts for a document.

Values 1-10 were tested for each parameter in steps of 1. The varying performance of CTR with different λ_t parameters illustrate the flexibility that CTR permits in treating each type of evidence differently. For example, to score keyphrases based on global context alone, we can set the λ_t values corresponding to the other contexts to zero.

Other types of neighborhoods can be included into the CTR model by appropriately extending the edge weights (Eq. (1)). We experimented with including the global contexts of citing and cited documents in the citation network. However, these global contexts did not provide additional improvements over citing and cited contexts. We do not include these experiments due to space restrictions.

How well does citation network information aid in key phrase extraction for research papers? Figure 4 shows MRR plots comparing CTR with SingleRank (SR) and ExpandRank (ER) on different datasets. In SR, edges are added between candidate word nodes if they occur within a window of w words of each other in the target document. Textually-similar neighbors are also considered in the ER model. Thus, window size w is a parameter for both SR and ER whereas the number of textually-similar neighbors is a parameter for ER. For CTR experiments, we include all citation network neighbors that are available. For each dataset and model (SR, ER, and CTR), we tested parameter values in steps of 1 in the range 1-10 and chose the best-performing settings in the comparison experiments presented in Figure 4. The "best-performing setting" corresponds to the setting that gives the highest MRR at *average* k for that dataset.

As can be seen in Figure 4, CTR substantially outperforms both SR and ER, illustrating that the cited and citing contexts contain significant hints that aid keyphrase extraction in addition to a document's content or its textually-similar documents. CTR is able to harness this evidence successfully to obtain good improvements in the extraction performance. For instance, compared to SR that uses document content alone, using CTR that incorporates multiple contexts and best-performing λ_t values, we achieve MRR@*average* k improvements of 28.66% for UMD, 23.73% for KDD, and 12.20% for the WWW datasets.

Figure 4 also shows MRR plots comparing CTR models when: (1) all the contexts for a document are used, the full CTR model; (2) only cited contexts are used, denoted CTR(Ctd); and (3) only citing contexts are used, denoted CTR(Ctg). The motivation for this comparison was to determine how well the proposed model performs on newly-published research papers that often accumulate citations not immediately, but over a period of time. Thus, these papers usually lack neighbors that cite them in the citation network although the neighbors they cite are available. As can be seen in the figure, CTR(Ctd) generally performs slightly better than CTR(Ctg), with both of them substantially out-

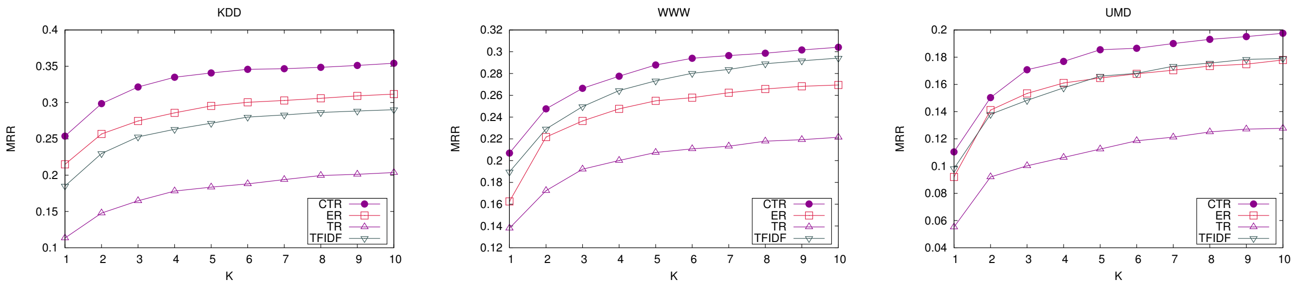

Figure 5: MRR curves for different keyphrase extraction methods. CiteTextRank (CTR) is compared with the baselines: TFIDF, TextRank (TR), and ExpandRank (ER).

1	COD: Online Temporal Clustering for Outbreak Detection (2007) {*outbreak detection, onset detection, detection task, epidemic outbreaks*} **Temporal Clustering, Outbreak Detection, Cluster Onset Detection, unsupervised learning**
2	Mixed Constraint Satisfaction: A Framework for Decision Problems under Incomplete Knowledge (1996) {*constraint satisfaction, knowledge constraint, mixed constraint, csp framework*} **Mixed Constraint satisfaction, CSP framework, incomplete knowledge, mixed CSP**
3	Recent Developments in NIKL (1986) {*knowledge representation, representation languages, kl-one family, nikl*} **KL-ONE, knowledge representation, NIKL**

Table 2: Sample predictions for AAAI papers using CiteTextRank. Human-picked keyphrases are shown in bold.

performing SR and ER. The full CTR model performs best among all compared models in the figure.

How does CiteTextRank compare with existing state-of-the-art methods? Figure 5 shows MRR plots comparing CTR with the baselines: TFIDF, TextRank (TR), and ExpandRank (ER). Based on recent studies, these baselines comprise state-of-the-art for keyphrase extraction (Mihalcea and Tarau 2004; Wan and Xiao 2008; Hasan and Ng 2010).

As the plots in Figure 5 indicate, CTR substantially outperforms other keyphrase extraction models. Similar results hold for precision and recall plots (results not shown due to space limitations). With a paired T-test, our improvements for MRR, precision and recall are statistically significant for p-values ≥ 0.05. Finally, we compare the best-performing baseline from previous research with CTR at *average* k in Table 3. CTR effectively out-performs the state-of-the-art baseline models for keyphrase extraction.

How does CiteTextRank perform on AAAI papers? Using CiteTextRank, we obtained precision and recall values of 22.8% and 27%, respectively, on the AAAI dataset along with an MRR value of 0.50 at $k = 4$. That is, on average, we are able to predict the gold-standard keyphrase at rank 2 among the top-4 predictions. We show some anecdotal examples in Table 2. The predictions obtained by CTR along with human-picked "gold" keyphrases are listed in this table. As can be seen, there is a high overlap between the "gold" and predicted keyword sets.

Dataset	Method	Precision	Recall	F1	MRR
UMD	BL*	0.0844	0.0956	0.0871	0.1610
	CTR	**0.0905**	0.0925	**0.0914**	**0.1769**
WWW	BL*	0.1000	0.1220	0.1085	0.2566
	CTR	**0.1099**	**0.1341**	**0.1192**	**0.2878**
KDD	BL*	0.1052	0.1219	0.1116	0.2858
	CTR	**0.1328**	**0.1529**	**0.1405**	**0.3348**

Table 3: The evaluation measures at *average* k are shown for the best baseline method (BL*) and CiteTextRank (CTR).

Conclusions and Future Directions

We addressed keyphrase extraction from scientific documents. In particular, we showed that in addition to the original textual content of a scientific document, the fact that the document is situated in an interlinked citation network can be effectively harnessed for extracting keyphrases. We proposed CiteTextRank (CTR), a flexible, unsupervised graph-based model for ranking keyphrases using multiple sources of evidence such as the textual content of a document, textually-similar neighbors and neighbors in the interlinked document citation network. Using CTR, we illustrated significant improvements over baseline models for multiple datasets of research papers in the Computer Science domain.

Unlike CTR, the baseline models are more general since they depend only on the document content and textually-similar neighbors. However, we believe that CTR will benefit scenarios where additional evidence is available from highly-connected networks that can help improve keyphrase extraction. Examples of such scenarios include papers from domains other than Computer Science, e.g., Biology, or images on the Web (where the anchor texts pointing to a webpage or to an image act as "citation contexts"). In future, we plan to extend and evaluate CTR for these other types of documents. Another interesting direction would be to extend CTR for extracting document summaries similar to (Mihalcea and Tarau 2004; Qazvinian, Radev, and Özgür 2010).

Acknowledgments

We are grateful to Dr. C. Lee Giles for the CiteSeerX data, which allowed the generation of citation graphs. We also thank Kishore Neppalli and Juan Fernández-Ramízer for their help with various dataset construction tasks. We very much appreciate the constructive feedback from our anonymous reviewers. This research was supported in part by the National Science Foundation award IIS-HCC 1353418 to Cornelia Caragea. Any opinions, findings, and conclusions expressed here are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

Abu-Jbara, A., and Radev, D. R. 2012. Reference scope identification in citing sentences. In *HLT-NAACL*.

Boudin, F. 2013. A comparison of centrality measures for graph-based keyphrase extraction. In *IJCNLP*.

Brin, S., and Page, L. 1998. The anatomy of a large-scale hypertextual web search engine. *Comput. Netw. ISDN Syst.* 30(1-7).

Frank, E.; Paynter, G. W.; Witten, I. H.; Gutwin, C.; and Nevill-Manning, C. G. 1999. Domain-specific keyphrase extraction. In *IJCAI*.

Giles, C. L.; Bollacker, K. D.; and Lawrence, S. 1998. Citeseer: An automatic citation indexing system. In *DL*.

Hammouda, K.; Matute, D.; and Kamel, M. 2005. Corephrase: Keyphrase extraction for document clustering. In *Machine Learning and Data Mining in Pattern Recognition*.

Hasan, K. S., and Ng, V. 2010. Conundrums in unsupervised keyphrase extraction: Making sense of the state-of-the-art. In *COLING (Posters)*.

Haveliwala, T.; Kamvar, S. K.; Klein, D.; Manning, C.; and Golub, G. G. 2003. Computing pagerank using power extrapolation. In *Stanford University Technical Report*.

Hulth, A. 2003. Improved automatic keyword extraction given more linguistic knowledge. *EMNLP* 216–223.

Jones, S., and Staveley, M. S. 1999. Phrasier: A system for interactive document retrieval using keyphrases. In *SIGIR*.

Kataria, S.; Mitra, P.; and Bhatia, S. 2010. Utilizing context in generative bayesian models for linked corpus. In *AAAI*.

Kim, S. N., and Kan, M.-Y. 2009. Re-examining automatic keyphrase extraction approaches in scientific articles. In *Proceedings of the Workshop on Multiword Expressions: Identification, Interpretation, Disambiguation and Applications*, MWE '09.

Kim, S. N.; Medelyan, O.; Kan, M.-Y.; and Baldwin, T. 2010. SemEval-2010 Task 5: Automatic Keyphrase Extraction from Scientific Articles. In *SemEval*.

Kim, S.; Medelyan, O.; Kan, M.-Y.; and Baldwin, T. 2013. Automatic keyphrase extraction from scientific articles. 47(3).

Liu, F.; Pennell, D.; Liu, F.; and Liu, Y. 2009. Unsupervised approaches for automatic keyword extraction using meeting transcripts. In *Proceedings of NAACL '09*, 620–628.

Liu, Z.; Huang, W.; Zheng, Y.; and Sun, M. 2010. Automatic keyphrase extraction via topic decomposition. In *EMNLP*.

Lu, Q., and Getoor, L. 2003. Link-based classification. In *ICML*.

Manning, C. D.; Raghavan, P.; and Schütze, H. 2008. *Introduction to Information Retrieval*. Cambridge University Press.

Marujo, L.; Ribeiro, R.; de Matos, D. M.; Neto, J. P.; Gershman, A.; and Carbonell, J. G. 2013. Key phrase extraction of lightly filtered broadcast news. *CoRR*.

Mihalcea, R., and Tarau, P. 2004. Textrank: Bringing order into text. In *EMNLP*.

Nguyen, T., and Kan, M.-Y. 2007. Keyphrase extraction in scientific publications. In *Asian Digital Libraries. Looking Back 10 Years and Forging New Frontiers*, volume 4822.

Page, L.; Brin, S.; Motwani, R.; and Winograd, T. 1999. The pagerank citation ranking: Bringing order to the web. Technical report.

Palshikar, G. K. 2007. Keyword extraction from a single document using centrality measures. In *International Conference on Pattern Recognition and Machine Intelligence*.

Pudota, N.; Dattolo, A.; Baruzzo, A.; Ferrara, F.; and Tasso, C. 2010. Automatic keyphrase extraction and ontology mining for content-based tag recommendation. *Int. J. Intell. Syst.*

Qazvinian, V.; Radev, D. R.; and Özgür, A. 2010. Citation summarization through keyphrase extraction. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, 895–903. Stroudsburg, PA, USA: Association for Computational Linguistics.

Rendle, S.; Balby Marinho, L.; Nanopoulos, A.; and Schmidt-Thieme, L. 2009. Learning optimal ranking with tensor factorization for tag recommendation. In *KDD*.

Ritchie, A. 2008. *Citation context analysis for information retrieval.* Ph.D. Dissertation, University of Cambridge.

Shi, X.; Leskovec, J.; and McFarland, D. A. 2010. Citing for high impact. In *JCDL*.

Song, Y.; Zhang, L.; and Giles, C. L. 2008. A sparse gaussian processes classification framework for fast tag suggestions. In *CIKM*.

Teufel, S. 1999. *Argumentative Zoning: Information Extraction from Scientific Text*. Ph.D. Dissertation, University of Edinburgh,.

Turney, P. D. 2000. Learning algorithms for keyphrase extraction. *Inf. Retr.* 2.

Wan, X., and Xiao, J. 2008. Single document keyphrase extraction using neighborhood knowledge. In *AAAI*.

Yin, D.; Xue, Z.; Hong, L.; and Davison, B. D. 2010. A probabilistic model for personalized tag prediction. In *KDD*.

Zha, H. 2002. Generic summarization and keyphrase extraction using mutual reinforcement principle and sentence clustering. In *SIGIR*.