A New Scheme for Scoring Phrases in Unsupervised Keyphrase Extraction

Corina Florescu and Cornelia Caragea

Computer Science and Engineering,
University of North Texas, Denton, TX, USA,
CorinaFlorescu@my.unt.edu; ccaragea@unt.edu;

Abstract. Many unsupervised methods for keyphrase extraction typically compute a score for each word in a document based on various measures such as *tf-idf* or the PageRank score computed from the word graph built from the text document. The final score of a candidate phrase is then calculated by summing up the scores of its constituent words. A potential problem with the sum up scoring scheme is that the length of a phrase highly impacts its score. To reduce this impact and extract keyphrases of varied lengths, we propose a new scheme for scoring phrases which calculates the final score using the average of the scores of individual words weighted by the frequency of the phrase in the document. We show experimentally that the unsupervised approaches that use this new scheme outperform their counterparts that use the sum up scheme to score phrases.

1 Introduction

Keyphrase extraction is the task of automatically extracting descriptive phrases or concepts that represent the main topics of a document. Keyphrases provide a concise description of the topics of a document and are particularly useful in many applications ranging from information search and retrieval [1,2] to document summarization [3,4], classification [5], clustering [6], and recommendation [7] or simply to contextual advertisement [8]. In this paper, we aim at improving scoring of candidate phrases in unsupervised approaches to keyphrase extraction, using research papers as a case study.

Unsupervised approaches to keyphrase extraction have started to attract significant attention recently since, unlike supervised approaches, they do not require large human-annotated corpora, which are often expensive or impractical to acquire. Unsupervised keyphrase extraction is formulated as a ranking problem, where each candidate word of a target document receives a score based on various measures such as *tf-idf* [9] or PageRank [10]. Candidate words that have contiguous positions in a document are then concatenated into phrases. To compute the score of a phrase, many existing unsupervised approaches typically *sum up* the scores of its constituent words [10,11], and the top-ranked phrases are returned as keyphrases for the document. A potential problem with the *sum up* scoring scheme is that the length of a phrase highly impacts its score, with longer phrases receiving a higher score. For example, let us consider a research paper that contains the phrase "matrix factorization model" and has "matrix factorization" as one of its gold-standard author-annotated keyphrases. After running an unsupervised algorithm called SingleRank [12] on the paper, we obtain the scores for the individual words "matrix," "factorization," and "model" as follows: 0.047, 0.042, and 0.054, respectively. Since these words are adjacent in text, and hence, form a phrase, by summing up their scores, we obtain a score of 0.143 for the phrase "matrix factorization

model," whereas the keyphrase "matrix factorization" receives a lower score of 0.089. We posit that the length of a phrase should not be the only factor that contributes to the *keyphraseness* of a phrase.

To reduce the impact of the length of a phrase on its score, we propose a new scheme for scoring phrases in unsupervised approaches. The new scheme uses *means*, e.g., the arithmetic or harmonic mean of the scores of its individual words, weighted by the frequency of the phrase in the document to quantify for the relevance of that phrase to the topics of the document. We incorporate this new scoring scheme into several representative unsupervised systems for keyphrase extraction and conduct experiments on three datasets of research papers. We show experimentally that the proposed scheme improves the performance of existing unsupervised approaches by as much as 76.28% (relative improvement in performance over current models).

2 Related work

The unsupervised methods for keyphrase extraction have received a lot of attention and are becoming competitive with supervised approaches [13,14]. The PageRank algorithm is widely-used in keyphrase extraction models. Other centrality measures such as betweenness and degree centrality were also studied for keyphrase extraction [15]. However, based on recent experiments, the PageRank family of methods and *tf-idf* ranking are considered state-of-the-art for unsupervised keyphrase extraction [14,16].

Mihalcea and Tarau [10] proposed TextRank for scoring keyphrases using the PageRank values obtained on a word graph built from the adjacent words in a document. Wan and Xiao [11] extended TextRank to SingleRank by adding weighted edges between words co-occurring within a window size greater than 2. Unlike TextRank and SingleRank, where only the content of the target document is used for keyphrase extraction, textually-similar documents are included in the ranking process in ExpandRank [11]. Gollapalli and Caragea [17] extended ExpandRank to integrate information from the citation network where papers cite one another. Other approaches leverage clustering techniques on word graphs to improve keyphrase extraction [18,19]. Liu et al. [20] proposed TopicalPageRank, which decomposes a document into multiple topics, using topic models, and applies a separate PageRank for each topic. The PageRank scores of each topic are then combined into a single score, using as weights the topic proportions returned by topic models.

Several other approaches *directly* rank phrases, instead of first ranking individual words and then aggregating their scores to rank phrases. For example, the best performing keyphrase extraction system in SemEval 2010 [21] used statistical observations such as term frequencies to filter out phrases that are unlikely to be keyphrases. More precisely, thresholding on the frequency of phrases is applied, where the thresholds are estimated from the data. The candidate phrases are then ranked using the *tf-idf* model in conjunction with a boosting factor which aims at reducing the bias towards single word terms. Danesh et al. [22] computed an initial weight for each phrase based on a combination of heuristics such as the *tf-idf* score and the first position of a phrase in a document. Phrases and their initial weights are then incorporated into a graph-based algorithm which produces the final ranking of keyphrases. Word embeddings are employed as well to measure the relatedness between words in graph based models [23].

In this work, we propose a new scoring scheme for models that compute the score of a phrase by *summing up* the significance scores of its constituent words in order to rank phrases. The proposed scheme averages the significance scores of constituent words in order to limit the contribution of the length of a phrase to its score.

3 Proposed Scoring Scheme

We propose to compute the score of a phrase using *mean*tf*, which corresponds to the mean of the scores of the individual words weighted by the frequency of the phrase within a document. The *mean* reduces the score of a phrase and confers importance to shorter phrases as well. Both arithmetic and harmonic mean can be used to score phrases. The *tf* component in *mean*tf* aims at increasing the score of phrases that occur frequently in a document.

Consider again the example phrase provided in the introduction, "matrix factorization model" and its word scores 0.047, 0.042, and 0.054, respectively. Computing the harmonic mean of the score of the words within the two phrases, we obtain a score of 0.047 for "matrix factorization model" and a score of 0.044 for "matrix factorization," making the longer phrase still more likely to be retuned as a keyphrase. However, by incorporating the frequency of the two phrases, we obtain a score of 0.132 for "matrix factorization," whereas the score of "matrix factorization model" remains 0.047. In general, if a 3-word phrase would be a keyphrase for the document, its frequency is expected to be high (similar to that of the 2-word phrase), and hence, our proposed scoring scheme would return the longer phrase as a keyphrase.

Hence, we propose to score a multi-word phrase p as: $R(p) = mean(p) * tf(p)$, where $mean(p)$ is the mean of the scores of individual words within the phrase p and *tf(p)* is the frequency of phrase *p* within the document. The *mean*tf* score is not a free-standing scoring scheme, but a step in unsupervised methods for keyphrase extraction. Therefore, we embed this scoring scheme into six well-known unsupervised algorithms, that first score words and then aggregate them to score phrases: Tf-Idf, TextRank, SingleRank, ExpandRank, CiteTextRank (CTR) and TopicalPageRank (TPR), which are briefly described below.

Tf-Idf [9]. In unsupervised methods for keyphrase extraction, *tf-idf* score is leveraged to rank candidate keyphrase. ***TextRank*** [10]. This method represents a document as a word graph according to adjacent words, then PageRank algorithm is used to measure the word importance within the document. ***SingleRank*** [11]. SingleRank extends TextRank adding weighted edges between words within a window of size greater than 2. ***ExpandRank*** [11]. In ExpandRank, textually-similar documents are included to enrich the knowledge in the word graph. ***CTR*** [17]. CTR extends ExpandRank by incorporating information from the citation network of a paper. ***TPR*** [20]. TPR runs multiple PageRanks on the word graph, one biased PageRank to each topic.

4 Experiments and Results

Datasets. We carried out experiments on three datasets of research papers. The first dataset was made available by Nguyen and Kan [24] and contains 211 research papers. The second and third datasets were made available by Gollapalli and Caragea [17] and consist of the proceedings of the ACM Conference on Knowledge Discovery and Data Mining (KDD) and the World Wide Web Conference (WWW), each with

834 and 1350 documents, respectively. In experiments, for all three datasets, we used the title and abstract of a research paper. The author-input keyphrases of a paper were used as gold-standard for evaluation. A summary of our datasets is provided in Table 1. For preprocessing, we used Porter Stemmer to reduce both extracted and gold-standard keyphrases to a base form. To train the topic model in TPR, we used $\approx 45,000$ papers extracted from the CiteSeer^x scholarly big dataset [25], compiled from the CiteSeer^x digital library. We evaluated the performance of the unsupervised models with *sum up*, *mean*, and *mean*tf* using the following metrics: Precision, Recall and F1-score, which are widely used in previous works [11,20]. We performed experiments using both harmonic and arithmetic mean, but no significant differences were found between the two means. Hence, we show results using the harmonic mean (*hmean*).

Dataset	#Docs	#Kp	#AvgKp	1-grams	2-grams	3-grams	n-grams ($n \geq 4$)
Nguyen	211	882	4.18	260	457	132	33
KDD	834	3093	3.70	810	1770	471	42
WWW	1350	6405	4.74	2254	3139	931	81

Table 1: A summary of our datasets.

Results and Discussion. Table 2 compares Precision, Recall and F1-score at top 5 predicted keyphrases for the six unsupervised methods using all three scoring schemes: *sum up* (baseline), *hmean*, and *hmean*tf*, on all three datasets, Nguyen, WWW, and KDD. Note that CTR was run only on KDD and WWW since citation networks are not available for Nguyen.

Unsupervised method	Nguyen			WWW			KDD		
	Precision	**Recall**	**F1-score**	**Precision**	**Recall**	**F1-score**	**Precision**	**Recall**	**F1-score**
Tf-Idf - sum up	.099	.128	.108	.099	.115	.103	.093	.116	.100
Tf-Idf - hmean	.122	.154	.133	.141	.155	.142	.119	.151	.129
Tf-Idf - hmean*tf	**.147**	**.184**	**.159**	**.161**	**.180**	**.164**	**.147**	**.186**	**.159**
TextRank - sum up	.087	.115	.097	.094	.110	.097	.086	.108	.093
TextRank - hmean	.091	.116	.100	.104	.116	.106	.086	.111	.094
TextRank - hmean*tf	**.112**	**.144**	**.123**	**.126**	**.142**	**.129**	**.117**	**.149**	**.127**
SingleRank - sum up	.079	.103	.087	.094	.109	.097	.093	.116	.100
SingleRank - hmean	.112	.139	.121	.137	.151	.138	.11	.137	.118
SingleRank - hmean*tf	**.136**	**.171**	**.147**	**.163**	**.182**	**.166**	**.150**	**.187**	**.162**
ExpandRank - sum up	.095	.121	.103	.111	.126	.114	.100	.129	.109
ExpandRank - hmean	.107	.141	.119	.139	.151	.140	.109	.143	.120
ExpandRank - hmean*tf	**.141**	**.183**	**.155**	**.165**	**.184**	**.168**	**.147**	**.189**	**.161**
CTR - sum up	-	-	-	.114	.132	.118	.107	.138	.117
CTR - hmean	-	-	-	.151	.166	.152	.127	.167	.139
CTR - hmean*tf	**-**	**-**	**-**	**.186**	**.209**	**.189**	**.173**	**.223**	**.190**
TPR - sum up	.077	.100	.084	.089	.113	.097	.089	.113	.097
TPR - hmean	.111	.137	.12	.113	.140	.121	.113	.140	.121
TPR - hmean*tf	**.134**	**.168**	**.145**	**.158**	**.198**	**.171**	**.149**	**.186**	**.161**
Tf - phrase frequency	.104	.129	.112	.132	.142	.133	.098	.125	.106

Table 2: Results of the comparison of various unsupervised models using *sum up*, *hmean* and *hmean*tf* to compute the compositional score of a phrase on three datasets, Nguyen, WWW, and KDD. The results are shown at top 5 predicted keyphrases. Best results are shown in **bold blue**.

As can be seen from the table, the models that use the aggregated score of a phrase based on *hmean*tf* substantially outperform their counterparts that use *sum up*. For example, on WWW, SingleRank with *hmean*tf* achieves an F1-score of 0.166 as compared with SingleRank with *sum up*, which achieves an F1-score of 0.097. Among all

unsupervised models, TPR and CTR achieve the highest improvement in performance by replacing *sum up* with *hmean*tf*, whereas TextRank has the lowest improvement. For example, on the WWW collection, the relative improvement in performance for CTR, TPR, and TextRank models is 60.16%, 76.28%, and 32.98%, respectively.

The models that use the aggregated score of a phrase based on un-weighted *hmean* also outperform the *sum up* baselines, for all datasets. For example, on Nguyen, ExpandRank with *hmean* has an F1-score of 0.119 as compared with 0.103 F1-score of ExpandRank with *sum up*. However, the models that use only *hmean* perform worse compared with their counterparts that use the weighted version *hmean*tf*. For example, on the same dataset, ExpandRank with *hmean*tf* reaches an F1-score of 0.155 as compared with 0.110 F1-score of ExpandRank with *hmean*. Thus, the frequency of a phrase acts as an important component in computing the aggregated score of a phrase for unsupervised keyphrase extraction. Note that, in supervised models, the frequency of a phrase (or its *tf-idf*) is one of the top-ranked features by Information Gain [26]. To better understand the benefit of associating the *hmean* and *tf* scores, we also compare *hmean*tf* with *Tf-phrase frequency*. *Tf-phrase frequency* calculates the score of both single and multi-word phrases based on their number of occurrences in the target document. As can be seen in Table 2, leveraging only the term frequency of a phrase yields worse performance compared with the aggregated score based on *hmean*tf*.

With a paired T-test, our improvements in the evaluation metrics are statistically significant for p-values ≤ 0.05.

5 Conclusion and Future Work

In this paper, we proposed a new scheme for scoring phrases in unsupervised keyphrase extraction, showing the benefits of emphasizing both one-word and multi-word phrases. Instead of using the *sum* to compute the aggregated score of a phrase (as is commonly done in the literature), we proposed the use of weighted *means* to compute these scores. The results of our experiments using the harmonic mean weighted by the phrase frequency, *hmean*tf*, show significant improvement in performance over the *sum* baseline on three datasets of research articles. Our findings can improve the performance of the keyphrase extraction task, which in turn, can improve indexing and retrieval of information in many application domains. In future, it would be interesting to explore the performance of *hmean*tf* on other types of datasets, e.g., news articles.

Acknowledgments. We very much thank our anonymous reviewers for their constructive comments and feedback. This research is supported by the NSF award #1423337.

References

1. Jones, S., Staveley, M.S.: Phrasier: A system for interactive document retrieval using keyphrases. In: Proceedings of the 22nd SIGIR. (1999) 160–167
2. Ritchie, A., Teufel, S., Robertson, S.: How to find better index terms through citations. In: Proceedings of the Workshop on How Can Computational Linguistics Improve Information Retrieval?, ACL (2006) 25–32
3. Zha, H.: Generic summarization and keyphrase extraction using mutual reinforcement principle and sentence clustering. In: Proceedings of the 25th SIGIR. (2002) 113–120

4. Qazvinian, V., Radev, D.R., Özgür, A.: Citation summarization through keyphrase extraction. In: Proceedings of the 23rd ACL. (2010) 895–903
5. Turney, P.D.: Coherent keyphrase extraction via web mining. In: Proceedings of the IJCAI. (2003) 434–442
6. Hammouda, K.M., Matute, D.N., Kamel, M.S.: Corephrase: Keyphrase extraction for document clustering. In: Machine Learning and Data Mining in Pattern Recognition. Springer (2005) 265–274
7. Pudota, N., Dattolo, A., Baruzzo, A., Ferrara, F., Tasso, C.: Automatic keyphrase extraction and ontology mining for content-based tag recommendation. International Journal of Intelligent Systems **25**(12) (2010) 1158–1186
8. Yih, W.t., Goodman, J., Carvalho, V.R.: Finding advertising keywords on web pages. In: Proceedings of the 15th WWW. (2006) 213–222
9. Zhang, Y., Milios, E., Zincir-Heywood, N.: A comparative study on key phrase extraction methods in automatic web site summarization. Journal of Digital Information Management **5**(5) (2007) 323
10. Mihalcea, R., Tarau, P.: Textrank: Bringing order into text. In: Proceedings of the EMNLP. (2004) 404–411
11. Wan, X., Xiao, J.: Single document keyphrase extraction using neighborhood knowledge. In: Proceedings of the 23th AAAI. (2008) 855–860
12. Wan, X., Xiao, J.: Single document keyphrase extraction using neighborhood knowledge. In: Proceedings of the 2008 AAAI. Volume 8. (2008) 855–860
13. Hasan, K.S., Ng, V.: Conundrums in unsupervised keyphrase extraction: making sense of the state-of-the-art. In: Proceedings of the 23rd ACL: Posters. (2010) 365–373
14. Hasan, K.S., Ng, V.: Automatic keyphrase extraction: A survey of the state of the art. In: Proceedings of the ACL. (2014) 1262–1273
15. Palshikar, G.K.: Keyword extraction from a single document using centrality measures. In: PReMI, Springer (2007) 503–510
16. Kim, S.N., Medelyan, O., Kan, M.Y., Baldwin, T.: Automatic keyphrase extraction from scientific articles. Language resources and evaluation **47**(3) (2013) 723–742
17. Gollapalli, S.D., Caragea, C.: Extracting keyphrases from research papers using citation networks. In: Proceedings of the AAAI. (2014) 1629–1635
18. Grineva, M., Grinev, M., Lizorkin, D.: Extracting key terms from noisy and multitheme documents. In: Proceedings of WWW. (2009) 661–670
19. Liu, Z., Li, P., Zheng, Y., Sun, M.: Clustering to find exemplar terms for keyphrase extraction. In: Proceedings of the 2009 EMNLP. (2009) 257–266
20. Liu, Z., Huang, W., Zheng, Y., Sun, M.: Automatic keyphrase extraction via topic decomposition. In: Proceedings of the EMNLP. (2010) 366–376
21. El-Beltagy, S.R., Rafea, A.: Kp-miner: Participation in semeval-2. In: Proceedings of the 5th international workshop on semantic evaluation, Association for Computational Linguistics (2010) 190–193
22. Danesh, S., Sumner, T., Martin, J.H.: Sgrank: Combining statistical and graphical methods to improve the state of the art in unsupervised keyphrase extraction. Lexical and Computational Semantics (2015) 117
23. Wang, R., Liu, W., McDonald, C.: Corpus-independent generic keyphrase extraction using word embedding vectors. In: Software Engineering Research Conference. (2014) 39
24. Nguyen, T.D., Kan, M.Y.: Keyphrase extraction in scientific publications. In: Asian Digital Libraries. Looking Back 10 Years and Forging New Frontiers, Springer (2007) 317–326
25. Caragea, C., Wu, J., Ciobanu, A., Williams, K., Fernandez-Ramirez, J., Chen, H.H., Wu, Z., Giles., C.L.: Citeseerx: A scholarly big dataset. In: ECIR. (2014)
26. Caragea, C., Bulgarov, F.A., Godea, A., Gollapalli, S.D.: Citation-enhanced keyphrase extraction from research papers: A supervised approach. In: EMNLP. (2014) 1435–1446