A Comparison of Supervised Keyphrase Extraction Models

Florin Bulgarov and Cornelia Caragea
Department of Computer Science and Engineering
University of North Texas
Denton, TX 76203
florinbulgarov@unt.edu, ccaragea@unt.edu

ABSTRACT

Keyphrases for a document provide a high-level topic description of the document. Given the number of documents growing exponentially on the Web in the past years, accurate methods for extracting keyphrases from such documents are greatly needed. In this study, we provide a comparison of existing supervised approaches to this task to determine the current best performing model. We use research articles on the Web as the case study.

Categories and Subject Descriptors

I.2.7 [**AI**]: Natural Language Processing

Keywords

Keyphrase extraction, citation contexts, supervised classifier

1. INTRODUCTION

Keyphrase extraction is the problem of automatically extracting important phrases or concepts from a document. Keyphrases provide a high-level topic description of a document and are rich sources of information for many applications such as document classification, clustering, recommendation, indexing, searching, and summarization. Due to the importance of keyphrases in many applications, a wide range of approaches to keyphrase extraction have been proposed in the literature along two lines of research: supervised and unsupervised. Graph-based algorithms and centrality measures are widely used in the unsupervised line of research. A word graph is built for each document such that nodes correspond to words and edges correspond to word association patterns. Nodes are then ranked using graph centrality measures such as PageRank and its variants [8, 9].

Different from this, in the supervised line of research, keyphrase extraction is formulated as a binary classification problem, where candidate phrases are classified as either positive (i.e., keyphrases) or negative (i.e., non-keyphrases) [2, 5]. Various feature sets and classification algorithms give rise to different models. For example, Hulth [5] used four different features in conjunction with a *bagging* technique. These features are: term frequency, collection frequency, the relative position of the first occurrence and the part-of-speech tag of a term. Frank et al. [2] developed a system called KEA that used only two features: *tf-idf* (term frequency-inverse document frequency) of a phrase and the *distance* of a phrase from the beginning of a document (i.e., its relative position) and used them as input to Naïve Bayes. Medelyan et al. [7] extended KEA to integrate information from Wikipedia and obtained improvements over KEA. Many of these approaches, both supervised and unsupervised, are compared and analyzed in a survey by Hasan and Ng [4]. However, most of these approaches consider only the textual content of a document or a document's local neighborhood, which is limited to textually-similar documents. In our recent work [3, 1], we showed that, in addition to a document's textual content and textually-similar neighbors, other informative neighborhoods exist that have the potential to improve keyphrase extraction. For example, in a scholarly domain, research papers are not isolated. Rather, they are highly inter-connected in giant *citation networks*, in which papers *cite* or *are cited* by other papers in appropriate contexts. These contexts are not arbitrary, but they serve as brief summaries of a cited paper.

Our citation context based approaches to keyphrase extraction [3, 1] outperform many existing unsupervised (e.g., TextRank [8] and ExpandRank [9]) and supervised (e.g., Hulth's [5] and KEA [2]) approaches. However, a comparison with the approach by Medelyan et al. [7], called Maui, was not performed. Maui shows best results in the supervised line of research. Hence, the goal of this study is to address this limitation of our previous work. That is, we show a comparison of our supervised citation context-based model, called CeKE [1], with the Maui supervised model [7]. In addition, we show improved performance of CeKE when adding one more feature into the model.

2. PRELIMINARIES

This paper aims at filling the experiments gap in the current literature in order to determine which automatic keyphrase extraction systems shows the best results. CeKE was introduced in [1] and combines features computed from the target paper, i.e. *tf-idf, tf-idf over a certain threshold, first position, relative position, first position under a threshold* and *part of speech*, as well as features extracted from the citation networks of research papers, i.e. *citation tf-idf* and two boolean features for determining if the phrase *appears in cited or citing contexts*. On the other hand, Maui [7] combines document-based features such as *tf-idf, relative position, keyphraseness, phrase length* and *spread* as

WWW 2015 Companion, May 18–22, 2015, Florence, Italy.
ACM 978-1-4503-3473-0/15/05.
http://dx.doi.org/10.1145/2740908.2742776.

	WWW			KDD		
Method	Precision	Recall	F1-score	Precision	Recall	F1-score
CeKE	0.228	0.386	0.285	0.213	0.413	0.280
Maui	0.120	**0.502**	0.193	0.104	**0.466**	0.170
Hulth - n-gram with tags	0.165	0.107	0.129	0.206	0.151	0.172
KEA	0.210	0.146	0.168	0.178	0.124	0.145
CeKE + keyphraseness - Naïve Bayes	**0.251**	0.460	**0.322**	**0.254**	0.440	**0.321**

Table 1: Results of our approach and its variations in comparison with current state-of-the-art systems.

well as externally-computed features that use Wikipedia as a source: *node degree, Wikipedia-based keyphraseness, semantic relatedness* and *inverse Wikipedia linkage*. The features used by Hulth's method and KEA were presented above.

Generating Candidate Phrases. To generate candidate phrases, we first apply part-of-speech filters using the NLP Stanford toolkit and then parse the title and abstract of each target paper, keeping only nouns and adjectives (similar to works in [1, 5, 6, 8, 9]). The remaining words are then stemmed using the Porter Stemmer, and those that had contiguous positions are merged into n-grams. An n-gram will have a maximum of 3 words. Finally, we eliminate the candidate phrases that end with an adjective (or unigrams that are adjectives), similar to the approaches used in [1, 9].

3. EXPERIMENTS AND RESULTS

Datasets. The datasets used in our experiments are made of research papers (titles, abstracts and citation contexts) from two top-tier machine learning conferences: World Wide Web (WWW) and Knowledge Discovery and Data Mining (KDD). Details of these datasets are provided in our previous works [3, 1]. For the evaluation phase, we used the author-annotated keyphrases obtained from the PDFs of the papers as our gold standard.

Results. We evaluated the models using the following metrics: precision, recall and F1-score, only for the positive class (*i.e. is keyphrase*). The reported values were averaged in 10-fold cross-validation experiments where the training and test sets were created at document level. As for the classifier, we used Naïve Bayes in all comparisons. The θ parameter was set to the (title and abstract) *tf-idf averaged* over the entire collection, while β was set to 20. These values were estimated on a validation set sampled from training, similarly to the approach in [1].

Table 1 shows the results of the comparison of our CeKE model (i.e., citation enhanced keyphrase extraction) with Maui, Hulth's and KEA. In addition, we show the results obtained with CeKE when we add the *keyphraseness* feature to its existing set of features. The *keyphraseness* feature shows how often a candidate phrase appears as a tag or a keyphrase in the training dataset. Hulth's implementation is based on the n-gram approach since this gives the best results among the 3 methods presented in the paper (see [5] for details). Moreover, this approach is the most similar to the candidate phrase generation used in all other methods. As can be seen from the table, *CeKE + keyphraseness* outperforms all other models in terms of precision and F1-Score. Although Maui achieves a relatively high recall of 0.502 for WWW and 0.466 for KDD, its overall performance is still significantly lower than that of CeKE and *CeKE + keyphraseness*. Maui has the lowest precision among all the tested methods. The simplest methods among all, i.e. KEA, manages to achieve a better performance than Hulth's model on WWW, but falls short on KDD where it has the lowest F1-score.

Our methods that make use of citation contexts, CeKE and *CeKE + keyphraseness*, have the highest and the most consistent results for both datasets. The reported values show that, adding the *keyphraseness* feature, improves both precision and recall of the CeKE method to an overall F1-score of 0.322 / 0.321 for WWW / KDD dataset, respectively, (in comparison with 0.285 / 0.280 - when the feature is not used). We experimented also by including features from Wikipedia into CeKE and into *CeKE + keyphraseness*, i.e., features such as those used in Maui, but the performance did not increased notably or it often decreased.

4. CONCLUSIONS

In this study, we provide a comparison of supervised approaches to keyphrase extraction. This comparison is meant to fill in the experiments gap currently existent in the literature and shows which model achieves the highest performance on the task of keyphrase extraction. We also show improvements in terms of performance for the CeKE algorithm by adding the keyphraseness feature to its feature set.

Acknowledgments

We are grateful to Dr. C. Lee Giles for the CiteSeerX data. This research was supported in part by the NSF award #1423337 to Cornelia Caragea.

5. REFERENCES

[1] C. Caragea, F. Bulgarov, A. Godea, and S. Das Gollapalli. Citation-enhanced keyphrase extraction from research papers: A supervised approach. In *EMNLP*, 2014.

[2] E. Frank, G. W. Paynter, I. H. Witten, C. Gutwin, and C. G. Nevill-Manning. Domain-specific keyphrase extraction. In *IJCAI '99*, 1999.

[3] S. D. Gollapalli and C. Caragea. Extracting keyphrases from research papers using citation networks. In *AAAI*, 2014.

[4] K. S. Hasan and V. Ng. Automatic keyphrase extraction: A survey of the state of the art. In *ACL*, 2014.

[5] A. Hulth. Improved Automatic Keyword Extraction Given More Linguistic Knowledge. In *EMNLP '03*, 2003.

[6] Z. Liu, W. Huang, Y. Zheng, and M. Sun. Automatic Keyphrase Extraction via Topic Decomposition. In *EMNLP '10*, pages 366–376, 2010.

[7] O. Medelyan, E. Frank, and I. H. Witten. Human-competitive tagging using automatic keyphrase extraction. In *EMNLP*, pages 1318–1327, Singapore, 2009.

[8] R. Mihalcea and P. Tarau. TextRank: Bringing Order into Texts. In *EMNLP 2004*, pages 404–411, 2004.

[9] X. Wan and J. Xiao. Single Document Keyphrase Extraction Using Neighborhood Knowledge. In *AAAI '08*, pages 855–860, 2008.