Available online at www.sciencedirect.com

ScienceDirect

Computer Speech & Language 47 (2018) 112–131

www.elsevier.com/locate/csl

RankUp: Enhancing graph-based keyphrase extraction methods with error-feedback propagation☆

Gerardo Figueroa, Po-Chi Chen, Yi-Shin Chen*

Department of Computer Science, Institute of Information Systems and Applications, National Tsing Hua University, Hsinchu 30013, Taiwan

Received 10 March 2016; received in revised form 30 January 2017; accepted 1 July 2017
Available online 20 July 2017

Abstract

In recent years, unsupervised, graph-based ranking algorithms have been successfully applied to keyphrase extraction tasks. These methods have the advantage of taking into account global information, such as text structure and relations between words, phrases, and sentences, rather than relying solely on local, vertex-specific information. Graph-based approaches for keyphrase extraction, however, have a particular drawback, which comes from their frequency-based analysis methods. The weakness is that many common, less relevant terms may get a higher ranking, particularly in short articles. The converse situation also occurs, where less common (and possibly more relevant) terms obtain lower rankings. We propose an unsupervised method—RankUp—that enhances graph-based keyphrase extraction approaches by applying an error-feedback mechanism similar to the concept of backpropagation. Experiments have been performed on almost 3,300 short texts from a variety of domains. Our experiments show that error-feedback propagation can boost the quality of keyphrases in graph-based keyphrase extraction techniques.

Keywords: Keyphrase extraction; Graph-based methods; Backpropagation; Error feedback; Short articles

1. Introduction

Keyphrases are highly condensed summaries that describe the contents of a document. They help readers know quickly what a document is about, and are generally assigned by the document's author or by a human indexer. However, with the massive growth of documents on the Web each day, it has become impractical to manually assign keyphrases to each document. The need for software applications that automatically assign keyphrases to documents has thus become necessary. Keyphrases have several other uses. In a single-document context, they can be used to replace the title of a document or to facilitate skimming when they are highlighted in the text. In a collection of texts, they can serve for indexing, document classification, clustering, and searching. Keyphrases can also facilitate search engine users with query refinement.

Traditional techniques often view keyphrase extraction as a classification task, in which candidate phrases are classified as either keyphrases or non-keyphrases. In these approaches, each candidate keyphrase is viewed as an independent phrase; that is, phrases do not influence each other. In recent years, graph-based ranking systems like

☆ This paper has been recommended for acceptance by Pascale Fung.

* Corresponding author.
E-mail address: gerardo_ofc@yahoo.com (G. Figueroa), wallat@gmail.com (P.-C. Chen), yishin@gmail.com (Y.-S. Chen).

http://dx.doi.org/10.1016/j.csl.2017.07.004
0885-2308/

Google's PageRank (Brin and Page, 1998) and HITS (Rose et al., 2010) have been successfully applied to keyphrase extraction tasks. Graph-based keyphrase extraction is different from traditional techniques. The key idea of these systems is to do "voting" or "recommendation", where the importance (score) of a term will contribute to its neighboring nodes. As a result, one node will influence all nodes in the graph. This approach takes into account global information, such as the article structure, rather than relying only on information related to an individual term.

Graph-based extraction has the drawback, however, that common or frequently used terms get higher scores because they have more edges connected to them. Similarly, rare or infrequent terms get lower scores because they are less connected. When observing this weakness on graph-based approaches, we find that the concept of backpropagation can be employed to overcome it.

In backpropagation learning, when an error is encountered at the output layer, it is propagated backward by apportioning it to each unit according to the amount of the error and the unit responsible. It not only adjusts the weight of edges that connect to the node but also propagates the error throughout the whole network. The trick is to assess the blame for an error and divide it among the contributing weights (Russel and Norvig, 1995). Our method combines this concept with graph-based keyphrase extraction, as backpropagation can be applied to many kinds of networks (Bishop, 2006). If any node is found with an incorrect score, the score and error are used to modify all the edge weights of the graph, making the final result more accurate.

We propose a novel unsupervised method, *RankUp*, which applies backpropagation to enhance graph-based keyphrase extraction algorithms. In this work, RankUp has been applied to two well-known methods: TextRank (Mihalcea and Tarau, 2004) and RAKE (Rose et al., 2010). Experiments have been performed on a large variety of texts, including questions from Stack Exchange and research paper abstracts from different domains. The results of our experiments show how backpropagation can be utilized on graph-based keyphrase extraction methods to produce better-quality keyphrases.

The remainder of this paper is organized as follows: Section 2 reviews previous work on supervised and unsupervised methods for automatic keyphrase extraction. Section 3 describes the overall framework of RankUp. Section 4 presents the setup and results of our experiments. Finally, we provide conclusions and future work in Section 5.

2. Related work

Traditional work on the automatic selection of keyphrases was based for the most part on either keyphrase assignment or keyphrase extraction; both utilizing machine-learning strategies. Keyphrase assignment selects keyphrases from a preexisting controlled list of words and phrases and allocates them to a text that can be best described by them. Keyphrase extraction, on the other hand, extracts these terms from the text itself (i.e., the words and phrases must exist in the text contents). Recent work has further generalized these approaches and categorized them as either *supervised* or *unsupervised* (Hasan and Ng, 2014). Fig. 1 illustrates the different categorizations for keyphrase extraction methods.

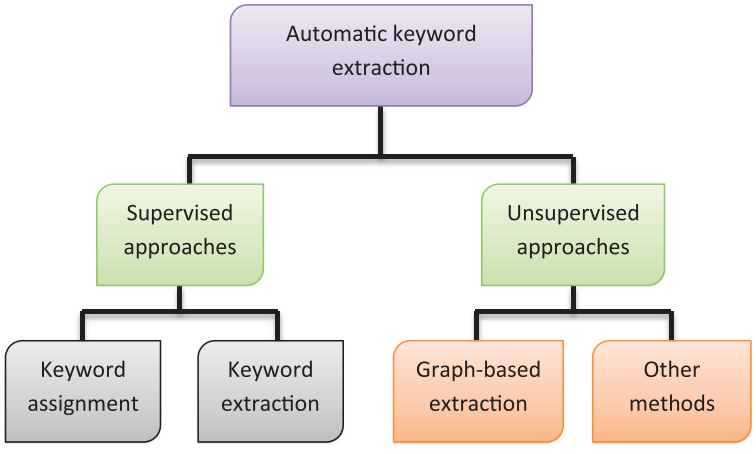

Fig. 1. Categorization of automatic keyphrase extraction methods.