Expert Systems With Applications 97 (2018) 51–59

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

A graph based keyword extraction model using collective node weight

Saroj Kr. Biswas*, Monali Bordoloi, Jacob Shreya

Computer Science and Engineering Department, National Institute of Technology Silchar, 788010 Assam, India

ARTICLE INFO

Article history:
Received 22 August 2017
Revised 12 December 2017
Accepted 13 December 2017
Available online 13 December 2017

Keywords:
Sentiment analysis
Keyword extraction
Graph based model
Centrality measure
Text mining

ABSTRACT

In the recent times, a huge amount of text is being generated for social purposes on twitter social networking site. Summarizing and analysing of twitter content is an important task as it benefits many applications such as information retrieval, automatic indexing, automatic classification, automatic clustering, automatic filtering etc. One of the most important tasks in analyzing tweets is automatic keyword extraction. There are some graph based approaches for keyword extraction which determine keywords only based on centrality measure. However, the importance of a keyword in twitter depends on various parameters such as frequency, centrality, position and strength of neighbors of the keyword. Therefore, this paper proposes a novel unsupervised graph based keyword extraction method called Keyword Extraction using Collective Node Weight (KECNW) which determines the importance of a keyword by collectively taking various influencing parameters. The KECNW is based on Node Edge rank centrality with node weight depending on various parameters. The model is validated with five datasets: Uri Attack, American Election, Harry Potter, IPL and Donald Trump. The result of KECMW is compared with three existing models. It is observed from the experimental results that the proposed method is far better than the others. The performances are shown in terms of precision, recall and F-measure.

1. Introduction

Keywords are defined as a series of one or more words which provide a compact representation of a document's content (Berry & Kogan, 2010; Boudin, 2013; Grineva, Grinev, & Lizorkin, 2009; Lahiri, Choudhury, & Caragea, 2014). Keywords are widely used to define queries within information retrieval (IR) systems as they are easy to define, revise, remember, and share. Other applications using keywords include automatic indexing, automatic summarization, automatic classification, automatic clustering, automatic topic detection and tracking, and automatic filtering (Palshikar, 2007). The task of mining these keywords from a document is called as keyword extraction. The manual assignment of keywords is a very time consuming and tedious task so it is important to have a proficient automated keyword extraction approach.

Micro-blogs have been recently attracting people to express their opinion and socialize with others. Micro blogging is a combination of blogging and instant messaging that allows users to create short messages to be posted and shared with an audience online. Social platforms like twitter have become extremely popular forms of this new type of blogging, especially on the mobile web – making it much more convenient to communicate with people compared to the days when desktop web browsing and interaction was the norm. Users share thoughts, links and pictures on Twitter, journalists comment on live events, and companies promote products and engage with customers. The list of different ways to use twitter could be really long, and with 500 millions of tweets per day, there is a lot of data to analyze and explore. One of the most important tasks in analyzing twitter data is keyword extraction. If keywords of a text are extracted properly, subject of the text can be studied and analyzed comprehensively and good decision can be made on the text.

Texts are commonly represented using the well-known Vector Space Model (VSM) (Salton, Yang, & Yu, 1975), however it results in sparse matrices to be dealt with computationally and while target application involves twitter contents, compared with traditional text collections, this problem becomes even worse. Due to the short texts (140 characters), diversity in twitter contents, informality, grammatical errors, buzzwords, slangs, and the speed with which real-time content is generated, an effective technique is required (Ediger et al., 2010) to extract useful keywords. Graph based technique to extract keywords is appropriate in such situation and has gained popularity in the recent times.

Bellaachia and Al-Dhelaan (2012) proposed a graph based method to extract keywords from twitter data, which uses node weight with TextRank and results in a node-edge weighting approach called NE-Rank (Node and Edge Rank). Term Frequency–Inverse Document Frequency (TF–IDF) is used as the node weight. But, keywords in twitter data do not only depend on TF–IDF.

* Corresponding author.
E-mail address: bissarojkum@yahoo.com (S.Kr. Biswas).

https://doi.org/10.1016/j.eswa.2017.12.025
0957-4174/

Abilhoa and Castro (2014) proposed a graph based technique to extract keywords from twitter data, which uses closeness and eccentricity centralities to determine node weight and, degree centrality as the tie breaker. Closeness and eccentricity centralities do not work well for disconnected graphs. However in most of the cases, the graph made from tweets becomes a disconnected graph due to the diversity of the tweet contents. Therefore, an effective graph based keyword extraction method is required which can overcome most of the drawbacks of graph based model including the ones cited above. This paper proposes such a graph based keyword extraction method called Keyword Extraction using Collective Node Weight (KECNW) which depends on many parameters of a node like frequency, centrality, position and strength of neighbors.

The remaining part of the research article is organized as follows. Section 2 presents literature survey which describes the related previous works. Section 3 discusses the proposed model in great detail. An illustrative example is presented in Section 4 to understand the proposed model clearly. Results with discussion are presented in Sections 5 and 6 draws some conclusions about the research work.

2. Literature survey

The keyword extraction techniques can be divided into four categories namely, linguistic approach, machine learning approach, statistical approach and other approaches (Zahang et al., 2008). Linguistic approach uses the linguistic properties of the words, sentences and documents and the most commonly examined linguistic properties are lexical, syntactic, semantic and discourse analysis (Cohen-Kerner, 2003; Hulth, 2003; Nguyen & Kan, 2007). Machine learning approach considers supervised or unsupervised learning for keyword extraction. Supervised machine learning approach induces a model which is trained on a set of known keywords and then is used to find the keywords for unknown documents (Medelyan & Witten, 2006; Witten, Paynter, Frank, Gutwin, & Nevill-Manning, 1999; Zhang, Xu, Tang, & Li, 2006). Statistical approach comprises simple methods which do not require the training data and are language and domain independent. The statistics of the words from document can be used to identify keywords such as n-gram statistics, word frequency, TF–IDF, word co-occurrences, PAT Tree etc. (Chen & Lin, 2010). Other approaches for keyword extraction in general combine all approaches mentioned above.

Graph based approach is a statistical approach. Recently some graph based methods for keyword extraction have been proposed. Litvak, Last, Aizenman, Gobits, and Kandel (2011) proposed an unsupervised, graph-based and cross-lingual key phrase extractor, known as DegExt which uses simple graph-based syntactic representation of text and web documents to enhance the traditional vector-space model by taking into account some structural document features. Bellaachia and Al-Dhelaan (2012) proposed a novel unsupervised graph based keyword ranking method, called NE-Rank which considers word weights in addition to edge weights when calculating the ranking. Bougouin, Boudin, and Daille (2013) proposed an unsupervised method that aims to extract key phrases from the most important topics of a document, called as TopicRank. Topics are defined as clusters of similar key phrase candidates. Beliga, Mestrovic, and Martincic-Ipsic (2015) proposed a node selectivity model for the task of keyword extraction. The node selectivity is defined as the average strength of the node. Abilhoa and Castro (2014) proposed a keyword extraction method from tweet collections that represents texts as graphs and applies centrality measures- degree, closeness and eccentricity, for finding the relevant vertices (keywords). Lahiri, Choudhury and Caragea (2014) experimented different centrality measures such as degree, strength, neighborhood size – order 1, coreness, pagerank etc. on word and noun phrase collocation networks for keyword extraction and analyzed their performance on four benchmark datasets. Kwon, Choi, and Lee (2015) proposed a model for term weighting and representative keyword extraction. Wang, Feng, and Li (2016) introduced Average Term Frequency (ATF) and Document Frequency (DF) to calculate the node weight. Martinez-Romo, Araujo, and Fernandez (2016) introduced an unsupervised algorithm for extracting key phrases from a collection of texts based on a semantic relationship graph, called SemGraph. Tixier, Malliaros, and Vazirgiannis (2016) introduced a new unsupervised keyword extraction technique that capitalizes on graph degeneracy. This technique applies the *K*-truss algorithm to the task of keyword extraction for the first time. Khan, Yukun, and Kim (2016) proposed a graph based re-ranking approach, called Term Ranker which extracts single-word and multi-word terms by using a statistical approach, identifies groups of semantically similar terms, estimates term similarity based on term embeddings and uses graph refinement and node centrality ranking. Xue, Qin, and Liu (2016) proposed a model to identify topics from folksonomy by using topic models. Ravinuthala and Reddy (2016) proposed a directed graph representation technique in which weighted edges are drawn between the words based on the theme of the document. Nagarajan, Nair, Aruna, and Puviarasan (2016) presented a keyword extraction algorithm where documents are represented as graphs, words of the documents are represented as nodes and the relation between the words of the documents is represented as edges. Then degree and closeness centrality measures are used for keyword extraction. Song, Go, Park, Park, and Kim (2017) proposed a method which considers three major factors that make it different from other keyword extraction methods. The three major factors are temporal history of the preceding utterances, topic relevance and the participants. The utterances spoken by the current speaker should be considered as more important than those spoken by other participants.

3. Proposed KECNW model

The KECNW model considers frequency, centrality, position and strength of neighbors of a node to calculate importance of the node. The implementation of the model is segregated in 4 phases: preprocessing, textual graph representation, node weight assignment and keyword extraction. The details of all the phases are given below.

3.1. Phase 1: pre-processing

Twitter is a micro-blog where people generally write in a conversational style. Tweets are known to be very noisy for any text mining task as they contain a number of symbols that do not have any useful information and make further processing ineffective. Therefore this model includes effective pre-processing phase which removes meaningless symbols from tweets and hence, effective keywords can be extracted. The steps for pre-processing are as follows:

i. *Remove username and retweet symbol:* Tweets often contain usernames beginning with the symbol '@'. Sometimes a tweet is also re-tweeted, which means a tweet by any user is shared again by other users and it contains the symbol RT. These usernames and retweet symbol do not contribute any significance to keyword extraction and act as noise. So, usernames and retweet symbols are removed.
ii. *Remove URLs:* Any URL links appearing in the tweets are removed as the model focuses only on the textual part of the tweet and URLs act as unnecessary noise while keywords are extracted.