ClRank: A Method for Keyword Extraction from Web pages using clustering and distribution of nouns

Mohammad Rezaei, Najlah Gali, Pasi Fränti
School of Computing
University of Eastern Finland
Joensuu, Finland
{rezaei, najlaa, franti}@cs.uef.fi

***Abstract*—text analysis of a web page is more difficult than the analysis of the text of normal document due to the presence of additional information, such as HTML structure, styling codes, irrelevant text, and presence of hyperlinks. In this paper, we propose an unsupervised method to extract keywords from a web page. The method extracts unigram nouns by applying part of speech tagging on the text. It then clusters the nouns based on their semantic similarity. It selects a number of keywords from the highest scored clusters. Experimental results show that our method outperforms state-of-the-art TextRank by 13 % in precision, 6 % in recall, and 10 % in F-measure.**

Keywords—web mining; keywords extraction; clustering; semantic analysis

I. INTRODUCTION

Keyword is the smallest unit that can express the meaning of a text. Keywords summarize the content of the document by few selected words [1]. They are easy to define by human, revise, remember and share. Keywords have been used in several tasks, such as information retrieval [2] document retrieval, document clustering [3], document classifying [4], indexing [5], summarization [6], and topic detection [7].

Documents such as scientific publications contain a list of keywords explicitly assigned by authors. However, most of other documents have no keywords assigned to them [8]. Manual assignment of keywords is labor intensive, time consuming and error prone. Several automatic keyword extraction methods have been proposed. These methods have been divided into four categories in [9]: statistical, linguistic, machine learning and other methods and into three categories in [10]: statistical, linguistic, and mixed methods. The latter categorization is more appropriate because machine learning methods are also based on statistical or linguistic knowledge to learn the model and it is not standalone category.

Normal text documents are often presented in one format such as title followed by abstract and main content. However, in web pages, the text is scattered over the page and the format differs from a category to another, which makes it more difficult to analyze its content. The web pages contain irrelevant text such as advertisements, formatting text such as navigation menus, styling codes such as java script (JS) and cascading style sheet (CSS), hyperlinks, and hypertext markup language (HTML) structure such as tags (see Fig. 1). In several cases, the size of this information is more than the size of the main text therefore; the task for keywords extraction is not trivial.

As reported in [11], [12], keywords that cover significant portion of a document are more important than keywords that cover a small portion. Existing methods have been mostly focused on judging the importance of words in isolation [11]. Less attention has been paid to the property of coverage the whole topics of the document.

In this paper, we propose a method to extract keywords from a web page by clustering unigram nouns based on their semantic similarity. The method extracts text nodes from the document object model (DOM) tree of the page and applies part-of-speech (POS) tagging to identify nouns. The nouns are lemmatized to their base form and a semantic similarity measure based on WordNet is applied between all combinations of pairs of unique lemmas. The nouns are clustered based on their similarity scores using hierarchical clustering.

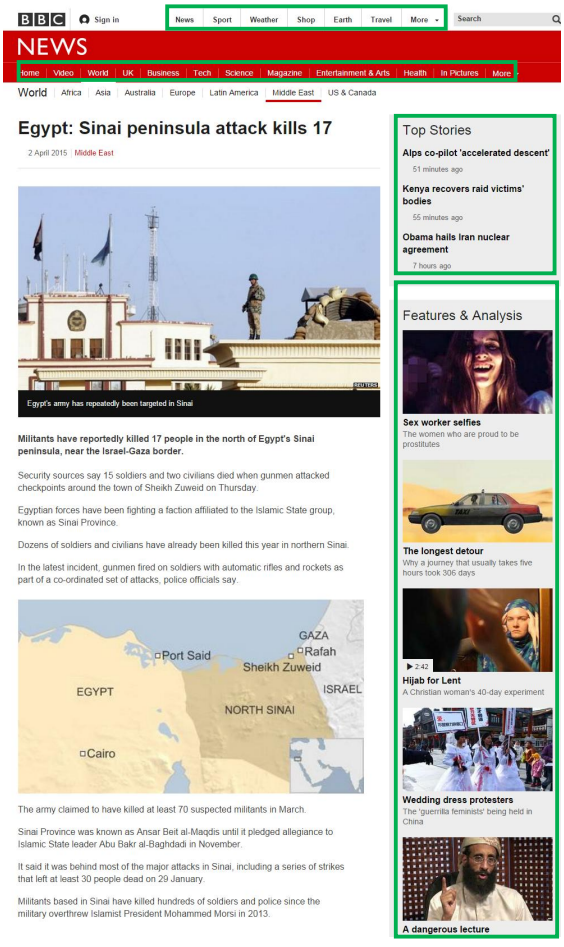

Figure 1. Example for a web page with irrelevant text.

The clusters are ranked based on the coverage of nouns to the page topics and the cluster size. Keywords are selected from the top ranked clusters.

The contribution of the paper is a new method for ranking the clusters that depends on the distribution of the nouns over the document. The goal is to extract keywords from the main text area with a full coverage to the page topics in the presence of irrelevant text such as short news articles in news page (see Fig. 1). The proposed method is unsupervised, domain independent, does not require corpus, and does not rely on HTML structure. We also study the effect of average-linkage, complete-linkage clustering, and human assigned keywords on the keyword extraction task.

II. KEYWORD EXTRACTION

Keyword extraction methods can be categorized into supervised or unsupervised approaches [13], [14]. Supervised approaches view keyword extraction as a classification task where each word in the document is either a keyword or not. A set of training data with labeled keywords is used to learn a model. The model is then applied to new set of documents to extract the keywords.

Genetic algorithm has been applied on a set heuristic rules to build the extractor [15]. Bayes' formula and two features (term frequency-inverse document frequency (TF-IDF), and the first appearance of the phrase in the document) have been used in [16] to build a Näive Bayes learning model. Results show that the performance is improved when using domain knowledge. A classification model is constructed in [8] using support vector machine (SVM). Five features are used: TF-IDF, first occurrence of the phrase in the document, position of occurrence, POS tag, and the relation dependency between words. SVM approach is also used in [17]. Supervised approaches lack in two aspects: first, they require training data with manually annotated keywords, which is not always available especially for web pages [12]. Second, they are bias toward the domain on which they are trained.

In unsupervised approaches a set of important keywords is selected from the document using various techniques such as clustering, graph based ranking and language modeling. TextRank [18] represents the document as a graph where the words are the vertices and the edges are the co-occurrence relation between the connected vertices within a specified size of window of words. The importance of each vertex is calculated using PageRank algorithm [19]. The words of the top ranked vertices are used to generate the keyphrases.

TopicRank [14] improves the work of [18] by clustering the noun phrases of the document using agglomerative clustering algorithm into topics. The document is represented as a complete graph where the vertices are the topics and the edges are the semantic relation between the connected vertices. TextRank ranking model is then applied to determine the importance of each topic and the keyphrases are selected from the top ranked topics.

SemanticRank [20] constructs a graph where words or sentences are the vertices and the edges are the semantic similarity measure between the vertices based on WordNet [21] and Wikipedia [22]. PageRank and hyperlink-induced topic search (HITS) [23] algorithms are applied for ranking, and the keyphrases are selected from the top ranked vertices. In general, graph based methods select the top ranked keyphrases, which may not guarantee a full coverage of the document topics [11].

In [24], clustering of noun phrases of a document has been proposed. Two noun phrases belong to the same cluster if they have one word in common. For example, two noun phrases *stem cell* and *stem cell research* are clustered together because they have *stem cell* in common. The clusters are scored by averaging the scores of the noun phrases in the cluster. The score of a noun phrase is calculated by the unigram frequency of the individual words in the noun phrase and the frequency of the noun phrase in the document. The shortest noun phrases from the highest scored clusters are selected as keywords. Using common words to cluster noun phrases will produce several small clusters that represent a same concept. For example, two words such as *machine* and *printer* will belong to different clusters because they do not have a word in common although they are semantically related. If the top clusters are semantically related then the extracted keywords will represent one topic and a full coverage to the document topics will not be achieved.

In [11], the words of a document are clustered based on their semantic relatedness and exemplar terms are obtained from the clustering. The exemplar terms are then used to extract noun phrases from the document as keyphrases. Two approaches to calculate the relatedness are considered: term co-occurrence and leveraging human knowledge. Wikipedia is adopted as the knowledge base to measure term relatedness. Three different clustering algorithms were tested: hierarchical, spectral and affinity propagation. Results have shown that Wikipedia-based relatedness provides slightly better results than word co-occurrences and spectral and hierarchal clustering outperform affinity propagation.

Spectral clustering is used in [12] to cluster the sentences of a document to find out the parts of text that are semantically related. Latent dirichlet allocation (LDA) is applied on the resulted clusters to discover the topics in each cluster. The keyphrases in the cluster are scored using a function that takes into account the distribution of the topics over the cluster, the distribution of the terms over the topics and the cluster size. The keyphrases with highest scores are selected to represent the document. Word co-occurrence is inefficient for one document or a small number of short texts as in web pages, because the co-occurrence matrix will be large and very sparse since most words do not co-occur with each other [25].

In [26], the keywords are extracted based on their relatedness weight among the entire text. The method uses term frequency to generate a list of candidate keywords. Word-to-word semantic similarity for all combination of pairs of words is calculated using adaptive lesk algorithm [27] and WordNet. The overall similarity (word-to-whole) is computed using similarity scores of all pairs of words. The importance of each keyword in the list is measured by dropping one word at a time and recalculating the overall similarity to see how it affects the cohesion of the keywords list. Negative result implies that the dropped word is important and a possible keyword for the document.

After applying different clustering approaches and a graph based method on several web pages, we found that these methods provide poor results in comparison with term frequency. Term frequency performs better and provides stable results after a pre-processing step such as removal of stop words. One reason is the heterogeneous structure of the web pages we studied; second, in most web pages, important words are emphasized by repeat; third, human tends to select words that appear several times in the page as keywords. However, there are web pages that use synonyms to emphasize the meaning and in such case term frequency fails.

To exploit the performance of term frequency in a method that supports all types of web pages, we use clustering. Our goal is to group high semantically related words such as *cost*, *price* and *charges* together so that each cluster represents a frequency of one concept. We use semantic relatedness between words based on WordNet to create the clusters. Semantic measure will overcome the problem of creating several clusters that represent one concept, while ranking the clusters based on the distribution of the nouns over the page will ensure a good coverage to the web page topics.

III. PROPOSED METHOD

There are six main steps involved in our method (see Fig. 2), which are as follows:

A. Preprocessing

We start by downloading the HTML source of the web page and parse it as a DOM tree. We do not use JS and CSS codes because their text content is mainly used for styling. Because of this, we remove script and styling tags from the tree. We use XPath[1], which is a query language for addressing parts of an XML document, to extract the text nodes from the tree (see Fig. 3)[2]. Symbols (&, £, $...) and numbers (1, 2, 3…) are removed from the text, after which the length of each text node is computed. If the length (number of unigrams) of the text node is less than 6 grams followed by a text node of a same length or less, then the text of the preceding node is deleted. For example, if a text node contains a unigram *home* followed by a text node contains two grams *shop online* then *home* is deleted and so on. However, if a text node contains (e.g. *Forme Spa*) followed by a text node that contains (e.g. *Forme Spa offers a tranquil environment designed for relaxation and rejuvenation*) then *Forme Spa* is not deleted. This preprocessing step ensures that the text of navigation menus, formatting and functional words is not considered as a part of the main text that we extract.

B. Part of Speech Tagging (POS)

When people do manually assign keywords, the majority of the selected words are either nouns or noun phrases [28]. Therefore, we extract unigram nouns as candidate keywords by applying POS tagging on the text fragments (see Fig. 3). POS assigns parts of speech such as noun, verb, and adjective to each word in the text based on its definition, and relationship with adjacent and related words in a phrase, sentence, or paragraph. In this paper, we use the tagger developed by Stanford University [29][3]. We also use a list of stop words to remove irrelevant words such as pronouns like *yesterday*, when they act as nouns.

C. Lemmatization

Lemmatization aims at removing inflectional endings of a word and return its base form, which is also known as lemma. All candidate nouns are lemmatized using Stanford lemmatization (see Fig. 3). Lemmatization is more robust than stemming as a pre-processing step when a similarity measure based on WordNet is used. The reason is that lemmatization involves usage of vocabulary and morphological analysis of words [30]. It returns the base form of words that are in dictionary. Stemming attempts to reduce a word to its stem by looking for prefixes or suffixes and remove them. It might fail and return words that have no meaning at all [31]. For example, the stem of *introduction* and *introduced* is *introduc* while the lemma is *introduce*. Lemmatization is also useful when counting the frequency of words in a document. For example, in lemmatization, the plural *mice* can be transformed to singular *mouse* but, the stem of *mice* is *mice*.

D. Similarity measure

We compute the semantic similarity between all pairs of unique lemmas using Wu and Pulmer measure [32], which is based on WordNet (see Fig. 3). If the lemma does not exist in WordNet, then all relevant nouns are removed from the list of candidates. Using lemmas instead of nouns for similarity computation will speed up the process because several nouns are associated with one lemma, therefore the generated similarity matrix is smaller and the similarity computation is faster.

E. Clustering

Several clustering methods exist such as partitional, hierarchical, grid-based and spectral; but the open question is which method performs best for keywords extraction task. Reference [11] reported a close performance for both spectral and hierarchical clustering. We use hierarchical clustering because the number of clusters can be controlled by simple thresholding. The nouns are clustered together using agglomerative algorithm if the similarity between their lemmas is greater than or equal to a threshold. In the experiment, we have tested average-linkage and complete-linkage with different threshold values. We continue hierarchical clustering until the similarity of the next clusters to be merged would be less than the threshold (see Fig. 3). The reason of using clustering is to group all relevant nouns together.

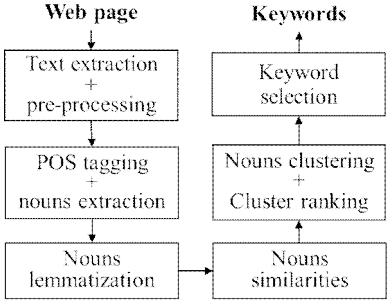

Figure 2. Keyword extraction algorithm.

[1] http://www.w3.org/TR/xpath20/

[2] http://www.formespa.co.nz/site/webpages/general/about-us

[3] http://nlp.stanford.edu/software/tagger.shtml

After all nouns have been assigned and the similarity between all clusters is less than the threshold, every cluster refers to a specific concept and the distribution of the nouns in each cluster over the whole page will reflect the importance of the cluster; where the more distribution of nouns over the page, the more coverage to the page topics.

The distribution is calculated by counting the number of text nodes in which a noun in the cluster appears. If the noun appears more than once in the same node, then it counts as one. For example, *Spa* appears 47 times in total, in 33 different text nodes (three times in the node shown in Fig. 3. The distribution counts of all nouns are summed up to represent the score of the cluster. In Fig. 3, the clusters are sorted in descending order based on their scores.

Ranking the clusters based on the distribution of nouns ensures that clusters that contain irrelevant nouns such as advertisement get low scores, because the nouns of the advertisement appear in a limited number of text nodes in one section of the page. We eliminate clusters that have small coverage over the web page by deleting any cluster of small size:

$$Size < 0.2 \times maxClusterSize \quad (1)$$

F. Selecting keywords

We use the frequency of the nouns in the web page as a criterion for selecting keywords from each cluster. We rank the nouns in each cluster based on their frequency in the page and we select the top frequent nouns (See Fig. 3). In case of tie, the average similarity to all other nouns in the same cluster is considered as a decision criterion. This favors nouns that are more central in the cluster. The number of keywords selected from each cluster depends on the frequency of the noun. The next keyword from the same cluster will be selected only if its frequency meets the following two conditions:

$$\begin{aligned} Frequency &> 0.2 \times maxFrequency \\ Frequency &> 3 \end{aligned} \quad (2)$$

Where *maxFrequency* is the maximum frequency found in the same cluster. The thresholds have been chosen empirically. The maximum number of keywords selected from each web page is limited to 10. In Fig. 3, one keyword is selected from every cluster, except for cluster 2, from which also noun *massage* is chosen. The method finds three (47%) correct keywords, misses four (57%), and selects three others that are not considered as correct choices by human.

IV. Experiments

A. Data Set

To our knowledge, previous works have used small set of 20, 23, 50 web pages in [33], [34], [26] respectively. We constructed a data set of 100 web pages divided into 5 different categories: *Education*, *News*, *Tourist*, *Beauty and fitness*, and *Food and drink*. All web pages in each category are from different sources of different types except for tourist category where the web pages follow the same template. The reason of having this variety in categories and heterogeneity in web pages is to see how the method performs in general.

Example of a text extraction and pre-processing

<h1>ABOUT FORME SPA</h1>
** **Forme Spa offers a tranquil environment designed for relaxation and rejuvenation. Whether it is pure INDULGENCE you are looking for with a Day Spa Package, RELAXATION with a massage from one of the far corners of the globe, REJUVENATION with a professional skin treatment/facial, or anti-ageing services such as microdermabrasion, skin peels, light therapy, IPL hair removal, or beauty therapy MAINTENANCE treatments, you've come to the right place. Our Heavenly Treatments are a selection of our top spa treatments for you to enjoy. ** **

POS tagging

Forme/NNP Spa/NNP offers/VBZ a/DT tranquil/JJ environment/NN designed/VBN for/IN relaxation/NN and/CC rejuvenation/NN ./. Whether/IN it/PRP is/VBZ pure/JJ INDULGENCE/NN you/PRP are/VBP looking/VBG for/IN with/IN a/DT Day/NNP Spa/NNP Package/NNP ,/, RELAXATION/NNP with/IN a/DT massage/NN from/IN one/CD of/IN the/DT far/JJ corners/NNS of/IN the/DT globe/NN ,/, REJUVENATION/NNP with/IN a/DT professional/JJ skin/NN treatment/facial/NN ,/, or/CC anti-ageing/JJ services/NNS such/JJ as/IN microdermabrasion/NN ,/, skin/NN peels/NNS ,/, light/JJ therapy/NN ,/, IPL/NNP hair/NN removal/NN ,/, or/CC beauty/NN therapy/NN MAINTENANCE/NNP treatments/NNP ,/, you/PRP 've/VBP come/VBN to/TO the/DT right/JJ place/NN ./. Our/PRP$ Heavenly/NNP Treatments/NNS are/VBP a/DT selection/NN of/IN our/PRP$ top/JJ spa/NN treatments/NNS for/IN you/PRP to/TO enjoy/VB ./.

Nouns extraction

Forme, Spa, environment, relaxation, rejuvenation, indulgence, day, package, massage, corners, globe, skin, treatment, facial, services, microdermabrasion, peel, light, therapy, IPL, hair, removal, beauty, maintenance, place, heavenly, selection

Nouns lemmatization

Forme, Spa, environment, relaxation, rejuvenation, indulgence, day, package, massage, corner, globe, skin, treatment, facial, service, microdermabrasion, peel, light, therapy, IPL, hair, removal, beauty, maintenance, place, heavenly, selection

Part of nouns similarities matrix

	Spa	**Building**	**Treatment**	**Massage**	**Therapy**
Spa	1.00	0.89	0.23	0.20	0.19
Building	0.89	1.00	0.70	0.67	0.64
Treatment	0.23	0.70	1.00	0.95	0.91
Massage	0.20	0.67	0.95	1.00	0.87
Therapy	0.19	0.64	0.91	0.87	1.00

Part of nouns clustering and clusters ranking

34	**Cluster 1:**	Spa (33)	Building (1)		
29	**Cluster 2:**	Treatment (20)	Massage (7)	Therapy (2)	
12	**Cluster 3:**	Albany (6)	Auckland (3)	City (2)	Gent (1)
8	**Cluster 4:**	Service (5)	Care (2)	Maintenance (1)	
7	**Cluster 5:**	Lady (4)	Hamilton (3)		

Keywords selection

Correct detection :	Spa	Treatment	Massage	
False detection:	Albany	Service	Lady	
Missed :	Relaxation	Beauty	Therapy	Facial

Figure 3. An example shows the output after each step in our algorithm, clustering method used is complete-linkage and merges threshold sets to 0.85.

The keywords were manually extracted by two students, so that for each web page we have two sets of keywords. After scanning through the selected keywords, we observed that the first one had selected more general keywords, while the second one had selected less but more precise keywords. The web pages with the labeled keywords are available for benchmarking upon request. In the rest of the paper we will refer to these labels as *set* 1 and *set* 2.

B. *Testing*

We conducted various experiments to investigate the influence of the merge threshold, clustering method, and human selection for the keywords. By changing the merge threshold, we can observe that both clustering methods provide stable values according to the F-measure at low thresholds (see Fig. 4 and 5).

A special case is when the threshold is set to 0.0. Then, all nouns are grouped into one cluster, and no cluster ranking will be used. The keywords are selected merely based on their frequency in the page, and in case of tie, the most central nouns in the cluster are selected. At this threshold, the method still performs better than TF and TextRank, and we conclude that it is an effective method for keyword extraction from web pages. The other special case is when the threshold is set to 1.0. In this case, only synonym nouns are grouped together (similarity must be 1.0) so that each cluster contains only one unique noun with its frequency. The ranking then merely depends on the distribution of the nouns over the page. At this threshold, we can observe that the method outperforms TF and TexRank, and performs better compared to if low threshold value was used. Form this we conclude that distribution of nouns over the document is more important than their frequency. This feature can be setup as criterion for selecting keywords rather than term frequency. Best F-measure was recorded at threshold 0.95, a situation where only highly similar nouns such as nouns and synonyms are grouped together.

We also compare the performance of the two clustering methods. Complete-linkage is more stable in respect to the selection of the merge threshold. Both clustering methods outperform TF and TexRank. Subjectivity of the human labeling has also a clear effect on the results. Highest precision 0.59 is recorded with *set* 1 because it contains more keywords than *set* 2. Best Recall 0.53 and F-measure 0.47 are recorded with *set* 2, because it contains more precise keywords. As shown in Fig. 4 and Fig. 5 human labels have also an impact on the behavior of the clustering methods. Average-linkage and complete-linkage behave opposite at low merge thresholds. Average-linkages outperforms complete-linkage until the merge threshold reaches 0.7 with *set* 2, while the performance of the clustering methods is close to each other with *set* 1.

C. *Comparison with other methods*

Table 1 shows the results of comparison with the state -of-the-art graph based method (TextRank) and term frequency (TF) after a pre-preprocessing step of removing stop words. We use a merge threshold of 0.95 for the comparison. We observed that average-linkage and complete-linkage outperform both TextRank and TF with *set* 1 and *set* 2. Table 2 shows the keywords extracted by each method. Same keywords are labeled in *set* 1 and *set* 2 for this specific web page.

V. CONCLUSION

We proposed an unsupervised method to extract keywords from web pages based on clustering and distribution of nouns over the page. We conducted various experiments using two sets of keywords for each web page that is manually extracted by humans. The results show that our method outperforms both state-of-the-art (TextRank) by 13 % in precision, 6 % in recall, and 10 % in F-measure and TF by 11 % in precision, and 6 % in F-measure. We conclude that clustering the nouns with the synonyms provided the best results. Distribution of nouns over the page is more effective feature than term frequency. Human selection for keywords has an obvious effect on the overall performance, where better F-measure results are achieved when human keywords are precise. Future work may focus on studying the effect of different similarity measures and clustering methods on keywords extraction from web pages.

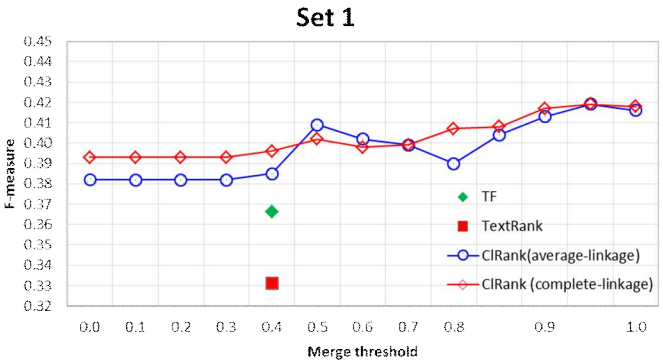

Figure 4. Average F-measure results with set 1.

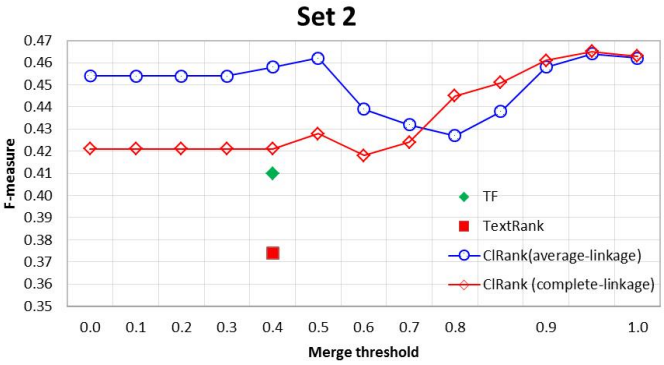

Figure 5. Average F-measure results with set 2.

TABLE I COMPARISON RESULTS OF TF, TEXTRANK, AND CLRANK

Method	Set 1			Set 2		
	P	***R***	***F***	***P***	***R***	***F***
TF	0.38	0.38	0.37	0.35	0.52	0.41
TextRank	0.36	0.33	0.33	0.33	0.46	0.37
ClRank (av.)	**0.51**	**0.38**	**0.42**	**0.46**	**0.52**	**0.46**
ClRank (comp)	**0.51**	**0.38**	**0.42**	**0.46**	**0.52**	**0.47**

TABLE II KEYWORDS EXTRACTED BY DIFFERENT METHODS

Method	Keywords
Ground truth	Spa, relaxation, massage, beauty, therapy, treatment, facial.
TF	Spa, forme, treatments, treatment, massage, Albany, skin, Auckland, Wellington, ngaio.
TextRank	Treatments, Spa, day, skin, I, time, massage, services, hours.
ClRank (av.) at 0.0	Spa, treatment.
ClRank (av.) at 0.95	Spa, treatment, massage
ClRank (av.) at 1.0	Spa, treatment.
ClRank (comp.) at 0.0	Spa, treatment, massage, eden, experience.
ClRank (comp.) at 0.95	Spa, treatment, massage
ClRank (comp.) at 1.0	Spa, treatment.

REFERENCES

[1] G. K. Palshikar, "Keyword extraction from a single document using centrality measures". In Pattern Recognition and Machine Intelligence. Springer Berlin Heidelberg 2007, pp. 503-510.

[2] X. Wan, J. Yang, and J. Xiao, "Towards an iterative reinforcement approach for simultaneous document summarization and keyword extraction". In Annual Meeting-Association for Computational Linguistics vol. 45, no. 1, p. 552, June 2007.

[3] S. S. Kang, "Keyword-based document clustering". In Proceedings of the sixth international workshop on Information retrieval with Asian languages, vol. 11, pp. 132-137. Association for Computational

[4] P. Tonella, F. Ricca, E. Pianta, and C. Girardi,. "Using keyword extraction for web site clustering". IEEE: In Web Site Evolution. Theme: Architecture. Proceedings, pp. 41-48, September, 2003.

[5] A. Gupta, A. Dixit, and A. K. Sharma, "A novel statistical and linguistic features based technique for keyword extraction". IEEE: In Information Systems and Computer Networks (ISCON), pp. 55-59, March 2014.

[6] E. D'Avanzo, B. Magnini, and A. Vallin, "Keyphrase extraction for summarization purposes: The LAKE system at DUC-2004". In Proceedings of the document understanding conference, May 2004.

[7] C. Wartena, and R. Brussee, "Topic detection by clustering keywords". IEEE: In Database and Expert Systems Application, DEXA'08, pp. 54-58, September, 2008.

[8] K. Zhang, H. Xu, J. Tang, and J. Li, "Keyword extraction using support vector machine". In Advances in Web-Age Information Management. Springer Berlin Heidelberg 2006, pp. 85-96.

[9] C. Zhang, "Automatic keyword extraction from documents using conditional random fields". Journal of Computational Information Systems, 4(3), 2008, pp. 1169-1180.

[10] M. J. Giarlo, "A comparative analysis of keyword extraction techniques". Rutgers, The state University of New Jersey 2005.

[11] Z. Liu, P. Li, Y. Zheng, and M. Sun, "Clustering to find exemplar terms for keyphrase extraction". In Proceedings of the Conference on Empirical Methods in Natural Language Processing: vol , pp. 257-266, Association for Computational Linguistics, August 2009.

[12] C. Pasquier, "Task 5: Single document keyphrase extraction using sentence clustering and Latent Dirichlet Allocation". In Proceedings of the 5th international workshop on semantic evaluation, pp. 154-157, Association for Computational Linguistics, July 2010.

[13] F. Boudin, "A comparison of centrality measures for graph-based keyphrase extraction". In International Joint Conference on Natural Language Processing (IJCNLP), pp. 834-838, October 2013.

[14] A. Bougouin, F. Boudin, and B. Daille, "Topicrank: Graph-based topic ranking for keyphrase extraction". In International Joint Conference on Natural Language Processing (IJCNLP), pp. 543-551, October 2013.

[15] P. Turney, "Learning to extract keyphrases from text", 1999

[16] E. Frank, G. W Paynter, I. H. Witten, C. Gutwin, and C. G. Nevill-Manning, " Domain-specific keyphrase extraction", 1999.

[17] C. Wu, M. Marchese, Y. Wang, M. Krapivin, C. Wang, X. Li, and Y. Liang, "Data preprocessing in SVM-based keywords extraction from scientific documents". IEEE: In Innovative Computing, Information and Control (ICICIC), pp. 810-813, December, 2009.

[18] R. Mihalcea, and P. Tarau, "TextRank: bringing order into texts". Association for Computational Linguistics, July 2004.

[19] S. Brin, and L. Page, "The anatomy of a large-scale hypertextual Web search engine". Computer networks and ISDN systems, 30(1), 1998, pp. 107-117.

[20] G. Tsattsaronis, I. Varlamis, K. Norvag, "SemanticRank: ranking keywords and sentences using semantic graphs". International Conference on Computational Linguistics, 2010, pp. 1074-1082.

[21] C. Fellbaum (Ed.). " WordNet: An electronic lexical database Cambridge", MA: MIT Press, 1998.

[22] D. N. Milne, , I. H. Witten, "An effective, low-cost measure of semantic relatedness obtained from Wikipedia links". In Proceeding of the 1st AAAI workshop on Wikipedia and Artificial Intelligence, 2008.

[23] J. M. Kleinberg, " Authoritative sources in a hyperlinked environment". Journal of the ACM (JACM), 46(5), 1999, pp.604-632.

[24] D. B. Bracewell, F. Ren, and S. Kuriowa, "Multilingual single document keyword extraction for information retrieval". In Natural Language Processing and Knowledge Engineering, IEEE NLP-KE'05, 2005, pp. 517-522.

[25] H. Coelho, "Classification of dreams using machine learning". In ECAI : 19th European Conference on Artificial Intelligence: Including Prestigious Applications of Artificial Intelligence (PAIS-2010): Proceedings , vol. 215, p. 169, IOS Press, August 2010.

[26] M. H. Haggag, "Keyword Extraction using Semantic Analysis". International Journal of Computer Applications, 61(1) 2013, pp.1-6.

[27] M. Lesk, "Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone". In Proceedings of the 5th annual international conference on Systems documentation ACM, pp. 24-26, June, 1986.

[29] K. Toutanova, D.Klein, C.Manning, and Y. Singer,"Feature-rich part-of-speech tagging with a cyclic dependency network. In Proceedings of HLT-NAACL 2003, pp. 252-259.

[28] A. Hulth, "Improved automatic keyword extraction given more linguistic knowledge." In Proceedings of the conference on Empirical methods in natural language processing, pp. 216-223. Association for Computational Linguistics, 2003.

[30] E. Al-Shammari, and J. Lin, "A novel Arabic lemmatization algorithm". In Proceedings of the second workshop on Analytics for noisy unstructured text data, ACM, pp. 113-118, July 2008.

[31] P. Han, S. Shen, D. Wang, and Y. Liu, "The influence of word normalization in English document clustering". IEEE International Conference in Computer Science and Automation Engineering (CSAE), vol. 2, pp. 116-120, May 2012.

[32] M. Pucher, F. T. W., "Performance evaluation of WordNet-based semantic relatedness measures for word prediction in conversational speech",2004

[33] Y. Zhang, N. Zincir-Heywood, and E. Milios, "World wide web site summarization", Technical Report CS-2002-8, Faculty of Computer Science, Dalhousie University, October 2002.

[34] R. Zarrad, N. Doggaz and E. Zagrouba, "Concepts extraction based on HTML documents structure". In ICAART (1), 2012, pp.503-506.