Keyword Extraction using Semantic Analysis

Mohamed H. Haggag
Computer Science Department
Faculty of Computers and Information, Helwan University

ABSTRACT
Keywords are list of significant words or terms that best present the document context in brief and relate to the textual context. Extraction models are categorized into either statistical, linguistic, machine learning or a combination of these approaches. This paper introduces a model for extracting keywords based on their relatedness weight among the entire text terms. Strength of terms relationship is evaluated by semantic similarity. Document terms are assigned a weighted metric based on the likeness of their meaning content. Terms that are strongly co-related to each other are highly considered in individual terms semantic similarity. Provision of the overall terms similarity is crucial for defining relevant keywords that most expressing the text in both frequency and weighted likeness. Keywords are recursively evaluated according to their cohesion to each other and to the document context. The proposed model showed enhanced precision and recall extraction values over other approaches.

Keywords
Keywords Extraction, Sematic Similarity, Semantic Relatedness, Semantic Analysis, Word Sense

1. INTRODUCTION
Keywords are useful for readers investigating in academic, business, social and other articles' purposes. They are used to give an insight of the presented article. They give a clue about the article concept so that readers can decide their interest. Their importance for search engines is to precise inquiry results and shortens the response time. Keywords can be viewed as content classifiers. Keywords processing tools are used for scanning large amount of text corpus in short times.

Extracting keywords from text is the process of addressing contextual representative terms. This includes access, discover and retrieval of different words and phrases highlighting linguistic senses and contents sub topics. Conceptually, cross-lingual keywords extraction could be a core and basic function for different natural language processing applications like text summarization, document clustering (unsupervised learning of document classes), document classification (supervised learning of document classes), topic detection and tracking, information retrieval and filtering, question answering and other text mining applications including search engines [25, 26 and 27].

Keywords extraction methodologies are classified into two main categories; quantitative and qualitative [1]. Quantitative techniques are based on set of concordances and statistical relations in addition to formal linguistic processing. Statistical approaches are the simplest keyword extraction models. Words statistics like word frequency, TF*IDF, co-occurrence are used to identify the keywords in the document. The basic intuition underlying this approach is that important concepts are more frequently referenced within the text than non-important ones. However, relying on simple counting of the most frequent words is not sufficient to address proper keywords. Accordingly, further refinements are needed to such approaches. Concordances are a set of words within the text along with their context reference. Statistical relations are advanced statistics that can be applied to data set resulted from the linguistic parsing of words and phrases.

Qualitative techniques are based on semantic relations and semantic analysis. Semantic analysis relies on semantic description of lexical items. The association of each lexical item with possible domains, in addition to words morphological and syntactic parsing, defines the semantic and conceptual relations of phrases. Extraction of keywords provides highly structured conceptual relations that relate to the content of the text. Such techniques are precise and reliable than plain statistical methods that results in important words but maybe unqualified.

Quantitative methods are widely used for a variety of text processing applications like text summarization, internal document cohesion, multi documents inter textual cohesion, authorship attribution, style consistency, writing skills development. On the other hand, the main application of qualitative techniques is content analysis.

Statistical keywords extraction from a single document presented in [2] starts by extracting frequent terms then extract a set of co-occurrences between each term and the frequent terms; i.e., occurrences within the same sentences. The Co-occurrence distribution shows the importance of a term in the document such that if the probability distribution of co-occurrence between any term and the frequent terms is biased to a particular subset of frequent terms, then the term is likely to be a keyword. The degree of bias of a distribution is measured by the $\chi 2$-measure.

Testing using text corpus, where documents are represented using a vector of features based on word co-occurrence was provided by [3]. A global unsupervised text feature selection approach based on frequent item set mining was applied. Each document is represented as a set of words that co-occur frequently in the given corpus of documents. Then documents are grouped to identify the important words using a locally adaptive clustering algorithm designed to estimate words relevance.

A different approach was used in [4], where a genetic algorithm for keyword extraction in a supervised environment was developed. According to [5], two extraction techniques were compared on a biological domain by extracting keywords from

MEDLINE that describe the most prominent functions of the genes. Resulting keywords weights are used as feature vectors for gene clustering. A comparison between two keyword weighting schemes: normalized z-score and (TF*IDF) is provided. Z-score refers to assigning a score to each word such that this score is based on the number of occurrences of the words in the document, the average occurrences in all the documents and the variance. The Z-score measure is similar to the standardization of a normal random variable. The best combination of background comparison set, stop list and stemming algorithm was selected based on precision and recall metrics.

A hybrid model that combines an artificial immune system with a mathematical background based on information theory was presented in [6]. This approach does not need neither domain knowledge nor the use of a stop words list. The output is a set of keywords for each of the corpus categories.

Many researches have been developed to enhance keywords extraction quality and precision by including domain linguistic knowledge. A combination of the evidence from frequency analysis and the hierarchically organized thesaurus was introduced by [7] using inductive logic programming. Linguistic knowledge was added by [8] to the representation, such as syntactic features, rather than relying only on statistics, such as term frequency and n-grams. In [9], three prediction models were combined to decide what words or sequences of words in the documents are suitable as keywords, the models were built using definitions of what constitutes a term in a written document. Existing state of the text-based information retrieval has been improved by automatic keyword extraction (AKE) [10] process for news characterization. Several linguistic techniques have been utilized. A keyword extraction technique that uses lexical chains was described .A lexical chain holds a set of semantically related words of a text. A lexical chain representing the semantic content of a portion of the text was showed in [11].

Web pages clustering are one of the text processing applications that are coherent to keywords extraction. Feature extraction known as Word Clusters was presented in automatic keyword extraction [10]. Word Clusters technique tried to cluster related words in the category they represent by using an algorithm named Bottleneck. This algorithm generates compact representations that improve the document processing. The generated clusters represent the main features of the document and also show an implicit relation between concepts.

Xinghua and Bin introduced automatic keyword extraction using position weights PW [13]. The word position plays an important role in linguistics. If the same word appears in the introduction and conclusion paragraphs of the document, the words generally carry more information. Also, words shown in the leading and summarization sentences are often more important than those in other positions of the same paragraph. Two words collections were used in this algorithm; transition phrase collection and summary phrase collection. PW is obtained by calculating the weights of each paragraph, sentence and word in the document. The algorithm was extended by applying semantic task.

A method for document categorization and the simultaneous generation of keywords for each category is presented in [14]. This method is based on K-Means, where each cluster is represented as a set of keywords. In this algorithm, each cluster has a set of features and a weight, from which the keywords for each category will be selected. Weights are adjusted in each iteration so that, at the end, they can define the most relevant keywords. A clustering method based on the automatic extraction of the keywords of a Web page was presented. The presence of common keywords was exploited to decide when it is appropriate to group pages together. A second usage of the keywords is in the automatic labeling of the recovered clusters of pages.

This paper proposes a new keyword extraction model based on semantic similarity. Semantic, or relatedness, similarity is utilized throughout the algorithm processing to dynamically evaluate the keywords cohesion as long as the words list change. The model starts with statistical keywords list construction as an initial set of keywords. Semantic similarity is then provided to this list to evaluate each word co-relation to the remaining words in the list. Words are dynamically removed while the other words cohesion is evaluated to decide which words need to be selected out and which will be kept as a final keywords significant to the document concept.

2. SEMANTIC ANALYSIS

Semantic similarity is the score of confidence which shows the two terms meanings relation semantically. High accuracy in semantic similarity evaluation is difficult to reach because the perfect semantic senses are only understood in a specific condition [24]. The concept of semantic similarity is different from semantic relatedness because semantic relatedness includes concepts like antonymy and meronymy while similarity doesn't. Semantic relatedness supports many relations such as Hypernym, Hyponym, Holonym, Meronym, Troponym, Antonym, Has-part, Part-of, Member-of and others. These relations are non-hierarchical WordNet Relations. For example, a leg is a part of a table, heavy is the opposite of light, snow is made up of water, and so forth. Semantic similarity represents a special case of semantic relatedness [28]. However, much of the literature uses these terms interchangeably, along with the term semantic distance.

Semantic similarity depends on only one relation between two concepts which measures the similarity degree between terms in the hierarchy. Some pairs of words are closer in meaning than others, for example car–tire are strongly related while car–tree are not. WordNet [16] is used to measure the similarity; it supports the similarity between noun pairs (e.g. cat and dog) and verb pairs (e.g., run and walk).

Lesk Algorithm, the Micheal Lesk algorithm [17], is used to disambiguate words in the sentence context. The adaptive Lesk algorithm enhanced the original algorithm by finding overlaps in WordNet glosses and Synsets of words to measure semantic relatedness rather than glosses found in traditional dictionaries.

3. PROPOSED MODEL

The proposed model starts with text preprocessing that enhance the extraction performance. Preprocessing starts with the removal of stop words, which are considered non-information bearing words like about, because, can, during,…. etc. This will to reduce noisy data. A list of 571 stop words has been used in this step. Following to stop words removal, stemming is provided to map all inflected words to their stems. The next step is non-WordNet words removal. As the proposed words similarity model use, but not limited to, the words synonym sets senses predefined in WordNet, the proposed model evaluation is properly addressed by limiting the words diversity to those covered by WordNet. However, a redirection to other lexicons could be provided for the case of non-WordNet words.

The statistical processing within the proposed model is provided by evaluating the Term frequency (TF), the number of times the word appears in a document divided by the number of total words in the document.

Semantic and relatedness similarity is the core process of the proposed model. Semantic similarity between extracted words refers to the relationship strength between the words. Processing the entire words, following to the above preprocessing, results in a set of all similarities between the document words. The most related keywords according to their semantic relations will be selected as per the following algorithm:

Initial Keywords List

The algorithm starts with a statistical extraction step based on the term frequency. Words, of number n, having the top TF are selected to be the n keywords list.

Semantic Relatedness

All possible combinations of all words in the keywords list are topologically organized in pairs. Each word of the keywords list appears in n-1 pairs along with other keywords, for example:

{Profit - sale}, {profit – revenue}, {profit - quarter}, {profit - internet}

Word pairs similarity

Evaluate word-to-word similarity using adaptive lesk algorithm for all words pairs. The output of this step represents the best sense for the two words in each pair along with their similarity score. An example of the output will be in the following format:

profit#n#1 sale#n#5 78

profit#n#1 revenue#n#1 19

profit#n#1 quarter#n#2 58

profit#n#1 internet#n#1 25

Similarity Score Normalization, word-to-word

The similarity score of each word pair (between words wi and wj) is normalized according to the following formula:

$$\text{Weighted similarity}(w_i, w_j) = (\text{similarity}(w_i, w_j) - \text{min similarity})/(\text{max similarity} - \text{min similarity}) \quad (1)$$

Average Similarity (word wi) & Average Similarity (n words), word-to-whole

The average weighted similarity score of each word (w_i) is calculated using the weighted scores of equation (1) for all pairs (w_i, w_1), (w_i, w_2), …….. (w_i, w_n).

The overall average weighted similarity score is then calculated for the whole n words w_1, w_2, …. w_n.

Words Similarity versus Cohesion

At this step, reverse similarity recursion is provided to validate the importance of words (one by one) to the overall similarity. The overall average weighted similarity is re-calculated for n-1 words by dropping one word in each word (w_i) score evaluation. In other words, weighted scores of equation (1) is calculated for the pairs (w_i, w_1), (w_i, w_2), …….. (w_i, w_{n-1}).

The effect of the dropped word is then evaluated by calculating the difference between the overall similarity on the basis of n and n-1 words similarity, as the following:

$$\Delta \text{ Overall Similarity} = \text{Overall Similarity}(n\text{-}1 \text{ word pairs}) - \text{Overall similarity}(n \text{ word pairs}) \quad (2)$$

check Δ Overall Similarity for the following options:

- Δ *Overall Similarity < 0*, fix to the words list is required.
- *0 < Δ Overall Similarity <= threshold (0.005*), the word is kept for further validation in the coming iterations.
- *0 < Δ Overall Similarity > threshold (0.005)*, the word is removed from the keywords list.

Negative difference in the first option means that the updated contribution, by removing on of the keywords list, will negatively affect the cohesion of the keywords list. Accordingly, this word need not to be dropped and another word should be selected.

Positive similarity difference is then required. However, in the second option, a small difference below threshold gives the impression that this word has no significant effect to the keywords list cohesion. The decision is to be postponed to later step after the words list will be changed.

Positive similarity differences higher than the threshold ensures that the selected word is effectively increased the keywords cohesion.

Keywords list stability

The above phase of the algorithm is repeated until no significant effect for the overall similarity is shown in response to words dropping.

Getting the more cohesion keywords, the most related keywords are now saved in the keywords list. The best sense for each keyword is then defined by selecting the sense with the highest number of occurrence. In case more than one sense for the keyword shared the same number of occurrence, the sense with the highest similarity score will be selected.

4. EVALUATION AND DISCUSSION

In this section, the proposed extraction model is evaluated by a set of experiments. Two different evaluation methods are used, the first is using the precision and recall measures and the second is evaluating the proposed model concept through one of the text processing applications.

Evaluation of the proposed model, which is developed based on keywords co-related semantic similarity, is done against four keywords extraction methods:

Term Frequency method; keywords are defined according to their term frequency. Most frequent terms are considered highly expressing the text.

Semantic Term Frequency method; the best sense of each word, resulted from TF method, is evaluated using Adaptive Lesk algorithm to the first occurrence of keyword in the document. A context window of size value 3 is used. Getting the best senses from all documents, the vector space model is constructed that includes semantic keywords.

Position Weight method; keywords are extracted according to their positions within the sentence and paragraph [13].

Semantic Position Weight method; Adaptive Lesk algorithm is applied to the position weight outcomes.

Experiments are done by assigning the same number of keywords extracted by the proposed model to the other four methods. For example, if the number of extracted keywords from a given document was 14 using the proposed model using co-related semantic similarity, the other four methods are tested for 14 keywords extraction as well.

4.1 Precision and Recall

Experimental results are conducted and analyzed using the 20 newsgroups data set, a collection of approximately 20,000 newsgroup documents partitioned evenly across 20 different newsgroups. Among them, five categories are used in the provided experiments; Windows, Graphics, Autos, Electronic and Politics-Mid east. Ten documents are selected from each category, where each has its predefined keywords.

Evaluation metrics considered are the Precision and Recall, which are the standard metrics for retrieval effectiveness in information retrieval. Basically calculated as follows:

$$Precision = TP / (TP + FP)$$

$$Recall = TP / (TP + FN)$$

Where:

TP= keywords extracted keywords by the algorithm and already found in document's predefined keywords.

FP= keywords extracted keywords by the algorithm and doesn’t found in document's predefined keywords.

FN= document's predefined keywords that are not extracted by the algorithm.

Table 1. Precision and Recall results

	Term Frequency	**Position Weight**	**Proposed Model**
Precision	77.2	80.3	83.5
Recall	68.5	73.0	76.3

Table 1 shows the precision and recall results for term keywords extraction based on term frequency method, position weight method and the proposed model. Proposed model results in enhanced precision and recall evaluation over other methods.

4.2 Keywords Extraction base for Applications

As introduced earlier, keywords extraction is considered core for many text processing and information retrieval applications. Throughout the proposed model evaluation, application of keywords extraction for documents clustering is provided. Document clustering is the process of finding out groups of information from documents domain and cluster these documents into the most relevant groups. Document clustering algorithms select specific document features and evaluate their matching against each document being clustered. For evaluation purposes, the effectiveness of the proposed model, as well as the other four keywords extraction methods, is provided by considering the extracted keywords as features of the documents vector space in the process of document clustering.

Documents clustering based on the proposed extraction model is achieved using CLUTO [21], a software package used for clustering low and high dimensional datasets and analyzing the characteristics of the various clusters. Bisection K-Means was used as clustering algorithm [22].

Entropy and purity are two common clustering quality metrics [23]. In the provided experiments, the two metrics were used to measure the quality of the documents clusters resulted from different algorithms. Entropy measures the classes of objects distribution within each cluster. Good clustering solutions result from small entropy values. Given a set of labeled documents belonging to I classes, assume the clustering algorithm partitions them into J clusters. Let n be the size of the document set; ni be the size of class i; nj be the size of cluster j; and nij be the number of documents belonging to both class i and cluster j. Then for a document in cluster j, the probability that it belongs to class i is P (i , j) = nij /nj.

The entropy of cluster j is:

$$E(j) = -\sum_{i=1}^{I} P(i,j)\log_2 P(i,j)$$

The entropy of the entire clusters is the sum of the entropy of each of the clusters weighted by its size:

$$E = \sum_{j=1}^{J} \frac{n_j}{n} E(j)$$

Purity measures how far each cluster contained objects from primarily one class. good clustering solution result from the large purity values. Similar to the entropy, the purity of each cluster is calculated as :

$$P(S_r) = \frac{1}{n_r} \max_i (n_r^i)$$

Where Sr is a particular cluster of size nr.

The purity of the entire clusters is computed as a weighted sum of the individual cluster purities and is defined as

$$Purity = \sum_{r=1}^{k} \frac{n_r}{n} P(S_r)$$

Documents clustering experiments are conducted using the proposed semantic similarity keywords extraction method and other four keywords extraction methods (Term Frequency, Word

Position, Semantic Term Frequency and Semantic Word Position). The five methods are evaluated and tested based on the Entropy and Purity of each clustering results. Experiments applied to four datasets of the 20 newsgroups. The four data sets increase in size linearly from DS1 to DS4. Results are shown in the following two tables.

Table 2. Entropy Results

Datasets	Term Frequency	Word Position	Semantic TF	Semantic WP	Proposed Model
DS1	0.324	0.319	0.312	0.305	0.239
DS2	0.326	0.325	0.313	0.3	0.229
DS3	0.324	0.315	0.313	0.295	0.218
DS4	0.322	0.311	0.311	0.29	0.17

Table 3. Purity Results

Datasets	Term Frequency	Word Position	Semantic TF	Semantic WP	Proposed Model
DS1	0.864	0.868	0.887	0.889	0.909
DS2	0.863	0.866	0.889	0.9	0.917
DS3	0.864	0.869	0.889	0.908	0.922
DS4	0.865	0.889	0.89	0.906	0.94

Entropy and purity results proved that the document clustering using the proposed keywords extraction model has the best quality over the other four methods for all data sets. It is clear from the above figures that purity values for term frequency and position weight methods don't increase as the data set size increase. However, applying semantic processing enhanced both term frequency and position weight methods in all data sets. In addition, efficiency of document clustering using semantic similarity increases linearly as the size of the documents dataset increase. The highest purity for DS1 is 0.909 and for DS4 is 0.94.

5. CONCLUSION

Keywords Extraction is widely used in text refinement. It is important in text processes such as clustering, classification or information retrieval. Keywords should express the core content of the document. This paper proposed an extraction model that utilizes semantic similarity features of the entire text items. The algorithm evaluates semantic relationships between individual words (word-to-word) and accordingly an overall similarity (word-to-whole) is scored. The extraction process undergoes semantic analysis recursion to properly define the keywords that are cohesion to the content. Results ensure enhanced precision and recall compared to traditional statistical and semantic approaches. The proposed model proved fitting capabilities to document processing applications. Clustering experiments based on the proposed keywords extraction approach showed better entropy and purity results compared to other methods.

6. REFERENCES

[1] Hunyadi, L. - Keyword extraction: aims and ways today and tomorrow. - In: Proceedings of the Keyword Project: Unlocking Content through Computational Linguistics. 2001.

[2] Y Matsuo, M Ishizuka, - Keyword Extraction from a Single Document Using Word Co-Occurrence Statistical Information- - International Journal on Artificial Intelligence Tools, 2004.

[3] N. Kang, C. Domeniconi, and D. Barbará. - Categorization and keyword identification of unlabeled documents. - In ICDM, pages 677–680. IEEE Computer Society, 2005.

[4] P. D. Turney. - Learning algorithms for key phrase extraction - . Information Retrieval, Springer, 2000.

[5] Y. Liu, B. J. Ciliax, K. Borges, V. Dasigi, A. Ram, S. B. Navathe, and R. Dingledine. - Comparison of two schemes for automatic keyword extraction from MEDLINE for functional gene clustering. - In CSB, pages 394–404. IEEE Computer Society, 2004.

[6] Andres Romero, Fernando Nino, - Keyword extraction using an artificial immune system, - Genetic And Evolutionary Computation Conference, Proceedings of the 9th annual conference on Genetic and evolutionary computation 2007.

[7] A Hulth, J Karlgren, A Jonsson, H Bostrom, L Asker – Automatic Keyword Extraction Using Domain Knowledge, Computational Linguistics and Intelligent Text Processing, 2001 – Springer.

[8] Anette Hulth. 2003a. - Improved automatic keyword extraction given more linguistic knowledge - . In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2003), pages 216–223, Sapporo, Japan. Association for Computational Linguistics.

[9] Hulth, A. (2004). - Enhancing Linguistically Oriented Automatic Keyword Extraction. - In Proceedings of the human language technology conference/North American chapter of the Association for Computational Linguistics Annual Meeting (HLT/NAACL 2004), Boston, May 2004.

[10] Martinez-Fernandez, J.L., A. García-Serrano, P. Martínez, and J. Villena, - Automatic Keyword Extraction for News Finder. - LNCS, 2004. 3094.

[11] Gonec Ercan, Llyas Cicekli- - Using lexical chains for keyword extraction - Information Processing and Management: an International Journal, ACM, Volume 43 , Issue 6 (November 2007).

[12] R. Bekkerman, R. El-Yaniv, N. Tishby, and Y. Winter. - Distributional word clusters vs. words for text

categorization. - Journal of Machine Learning Research, 3:1183–1208, 2003.

[13] Xinghua Hu; Bin Wu - Automatic Keyword Extraction Using Linguistic Features,- Data Mining Workshops, 2006. ICDM Workshops 2006. Sixth IEEE International Conference, Page(s):19 - 23 Dec. (2006)

[14] H. Frigui and O. Nasraoui. Simultaneous categorization of text documents and identification of cluster-dependent keywords, Apr. 07 2002.

[15] P. Tonella, F. Ricca, E. Pianta, and C. Girardi. Using keyword extraction for web site clustering. In WSE, pages 41–48. IEEE Computer Society, 2003.

[16] C. Fellbaum (Ed.), WordNet: An electronic lexical database, MIT Press, 1998.

[17] M. Lesk, - Automatic sense disambiguation using machine readable dictionaries: how to tell a pine code from an ice cream cone - in: Proceedings of the 5th annual international conference on Systems documentation, ACM Press, 1986.

[18] Ted Pedersen, Siddharth Patwardhan, and Jason Michelizzi. WordNet::Similarity – measuring the relatedness of concepts. In Proceedings of the Fifth Annual Meeting of the North American Chapter of the Association for Computational Linguistics, Boston, Massachusetts, 2004.

[19] Ted Pedersen, Satanjeev Banerjee, and Siddharth Patwardhan. Maximizing semantic relatedness to perform word sense disambiguation. Technical Report UMSI 2005/25, University of Minnesota Supercomputing Institute, March 2005.

[20] BBC dataset, Machine Learning Group; http://mlg.ucd.ie/.

[21] CLUTO. A Clustering Toolkit. Release 2.1.http://www-users.cs.umn.edu-/ karypis/cluto.

[22] M. Steinbach, G. Karypis, and V. Kumar. A Comparison of Document Clustering Techniques , KDD Workshop on Text Mining, 2000.

[23] Zhao, Y., & Karypis, G. Criterion functions for document clustering: Experiments and analysis. Technical Report TR #01–40, Department of Computer Science, University of Minnesota, Minneapolis, MN. (2001).

[24] Junsheng Zhang, Yunchuan Sun, Huilin Wang, and Yanqing He. "Calculating Statistical Similarity between Sentences". Journal of Convergence Information Technology, Volume 6, Number 2. February 2011

[25] Peter Turney, "Extraction of Keyphrases from Text: Evaluation of Four Algorithms", National Research Council of Canada, Canada, October 23, 1997

[26] J.Naveenkumar, "Keyword Extraction through Applying Rules of Association and Threshold Values", International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE), ISSN: 2278–1021, Vol. 1, Issue 5, pp 295-297, July 2012

[27] Jasmeen Kaur and Vishal Gupta, "Effective Approaches for Extraction of Keywords", International Journal of Computer Science Issues (IJCSI), ISSN (Online): 1694-0814, Vol. 7, Issue 6, pp 144-148, November 2010

[28] Philip Resnik, "Semantic Similarity in a Taxonomy: An Information-Based Measure and its Application to Problems of Ambiguity in Natural Language", Journal of Artificial Intelligence Research 11, pp 95-130, 1999