Arabic Keyword Extraction using SOM Neural Network

Ebtehal H.Omoush
Department of Computer Science
Al Albayt University
Mafraq, Jordan
hope_ebt@yahoo.com

Venus W. Samawi
Department of Computer Information System
Amman Arab University
Amman, Jordan
venus@aau.edu.jo

***Abstract*—Keywords are considered an abridged version of the text which indicate the important information implied within the document. The availability of huge amount of information on the WWW makes the process of analyzing document information and finding the proper keywords manually very difficult. Therefore, automatic keyword extraction techniques (AKE) are needed. In this paper, we will tackle the problem of automatic keyword extraction from Arabic documents base on unsupervised learning method. The main objective of this research is to propose an automatic Arabic keyword extraction (AAKE) technique from single document using full-text based indexing. The proper feature-set that improves AAKE performance is specified. Self-organizing map (SOM) neural network is used as an unsupervised learning method. The performance of the proposed technique is evaluated using recall, precision, and F-measure. Encouraging results are obtained compared with Sakhr keyword extractor.**

Keywords-Arabic keywords extraction; Self organized neural network; Natural language processing

I. Introduction

Nowadays, massive amount of text documents available on the WWW. To improve information retrieval process, documents should contain keyword list. Most old documents lack keyword list, which make the process of retrieving relevant documents (especially old documents) very difficult [1, 2]. Therefore, automatic keyword extraction techniques have emerged to improve text document retrieving process and locate documents that are most relevant to user queries [3, 4]. Keywords are set of words or phrases semantically covering most of the text within the document. Sometimes it is more proper to use multi-word (key-phrase) to search for a document. Key-phrases can be defined as "a list of terms each of which is made up of one or more words that describe the document with which they are associated" [5]. Keyword extraction is highly related to automated text summarization. In text summarization, most indicative sentences are extracted to represent the text [6], while in keyword extraction; most indicative words that signify the content of the document are extracted [1]. Keyword extraction is an important technique for document retrieval, Web page retrieval, document clustering, text mining, etc. It could be implemented manually or automatically. Manual keyword extraction is considered a burdensome task which requires intensive expert human effort. Therefore, it is important to construct a high performance AKE [7].

Few attempts have been made to develop an AKE from Arabic documents. The limited works on Arabic language is due to its complex nature [2, 6]. Based on the literatures, much research used TF (Term Frequency) as feature; perform stemming, and removing stop word as preprocessing steps. Researchers used different keyword identification method. Some of them used threshold, others used supervised systems (such as back-propagation, Multilayer Perceptron), in addition to statistical approaches (K Nearest Neighbor). In [7], a full-text based auto-indexing method for Arabic text documents is proposed. This method is mainly based on morphological analysis and on a technique for assigning weights to words depending term frequency (TF), the count of the stem words for that word, and the spread of that word over the document. The authors [5] suggested KP (key-phrase)-Miner system, which is capable of extracting key-phrases from both Arabic and English documents using heuristics rule. Stemming is applied as preprocessing step. They used TF, Inverted Document Frequency (IDF), boosting factor, and term position. A hybrid approach for automatic extraction of Arabic Multi-Word Terms (MWTs) using linguistic filter (TF is calculated for the word and its stem), and statistical filter (based on bigrams) is suggested in [8]. In [9], a supervised learning technique (Linear Discriminant Analysis) for key-phrases extraction from Arabic text documents is suggested, where linguistic knowledge and statistical information are used. Author in [10] applied a method to identify the keywords by combining linguistics and statistical analysis of the text document. Few researches utilized Artificial Neural Networks (ANN) for AKE [11, 12]. Most researchers used collect corpus from articles. Up to our knowledge, no attempts are made to specify the best features that help in keyword extraction for Arabic documents. Also, the ability of unsupervised techniques in AKE is not studied well. Consequently, we suggest unsupervised clustering approach using Kohonen's Self-organizing map (SOM) to perform automatic Arabic keyword extractor (AAKE) from single document using full-text indexing. Based on the clustering accuracy and retrieved results, the best feature-set will be indicated. Recall, precision, and F-measure are used to evaluate the performance of AAKE. The rest of this paper is organized in 5 sections. AKE is explained in section 2. Section 3 illustrates the main model and phases of (AAKE). System evaluation is shown in section 4. We concluded in section 5.

II. Automatic Keyword Extraction

To perform automatic keyword extraction, different approaches have been proposed. Based on [3, 14], the AKE methods are divided into four categories: statistical, linguistic, machine learning, and mixed approaches.

A. Statistics Approach

Statistical approach focuses on nonlinguistic features. The statistics features of the words can be used to identify the keywords in text. Many statistical features [13] could be used. In this work, 9 features are used, these are:

1) Term Frequency (TF): indicates the importance of a word to the document. Eq. (1) is used to find TF.

$$\mathrm{TF}(t,d) = \#\text{of times term t occurs in document d} \qquad (1)$$

First, represent text document as bag of words (BoWs), then find TF for each word (term) [2].

*2) TF ratio (RTF)***:** could be found using Eq. (2) [13].

$$RTF = \frac{TF}{|T_d|} \qquad (2)$$

TF is term frequency; the denominator indicates the total number of the words (terms) in that document d.

3) Title (T): Despite the word frequencies, the appearance of the word within the title is taken in consideration [15]. It is assigned as true or false (1 if the word appears in the title, else 0).

4) Sentence Frequency (SF): indicates the number of sentences that t occurs in [16].

5) SF ratio (RSF): could be found using Eq. (3) [13].

$$RSF = \frac{SF}{|S_d|} \qquad (3)$$

SF *is* sentence *frequency*; the denominator indicates the total number of the sentences in document d.

6) The important sentences always occur in specific positions: introduction and conclusion [15]. Therefore, occurrence of the word in the first and the last sentences are taken into consideration.

- *First Sentence (FS):* this feature is set to 1 if the word appears in the 1st sentence in the given document; else FS is set to 0.
- *Last Sentence (LS)*: Set to 1 if the word appears in the last sentence of the document; else set to 0.

7) As term frequency (TF) and sentence frequency (SF) amounts may differ depending on the document size. Therefore, to study the effect of normalizing frequency, the following features are used:

- Normalization Term Frequency (NTF): To normalize TF, Eq. (4) is applied [17].

$$NTf_i = \frac{(Tf_i - Tf_{\min})}{(Tf_{\max} - Tf_{\min})} \quad i = 1 \text{ to number of terms} \qquad (4)$$

- Normalization Sentence Frequency (NSF): we applied the equation of normalization [17] on the features of SF, as in Eq. (5).

$$NSf_i = \frac{(Sf_i - Sf_{\min})}{(Sf_{\max} - Sf_{\min})} \quad i = 1 \text{ to number of sentences} \qquad (5)$$

8) Linguistics Approach

Language description is based on morphology, syntax, and semantic analysis represents linguistic analysis [13]. The performance of linguistic analyzer requires language lexicon, morphology analyzer, and extraction of linguistic features of the word (such as class, gender, count, and person). In this work, only morphology (light stemmer) is applied. Light stemming is the process of eliminating prefixes and/or suffixes, but not the infixes.

B. Machine Learning Approach

Machine learning approach uses training *documents* to extract keywords, and produce AKE model. The produced model could be applied on any document to extract keywords. Keyword extraction has been treated as a classification task implemented using supervised and unsupervised machine learning approach. Machine learning approach includes Naïve Bayes, ANNs, Support Vector Machine (SVM), Neural Network etc. [12, 14].

C. Mixed Approach

AKE could be done by combining the methods mentioned above, or use some heuristic knowledge in the task of keyword extraction, such as position, length, layout feature of words, etc. [13,14].

III. Methodology of AAKE

In this research, automatic Arabic keyword extraction (AAKE) approach is developed to extract keywords from single Arabic document. Figure (1) summarizes the developed AAKE model. A mixed approach, where statistical, linguistics, and decision making based on machine learning approaches are used to construct AAKE. To extract keywords from a document, first, the document is tokenized (partition the document in to valid tokens) to produce set of tokens. The generated set of tokens is passed through AAKE phases to extract keywords. AAKE model mainly consists of three phases. *Preprocessing phase,* which performs normalization, filters the tokens, and apply light stemming. The preprocessing phase will generate index or word-list

(Bows) that includes the document’s words that are candidate to be keywords. The second phase is the *features extraction phase*. This phase is concerned with extracting statistical features of each word$_i$ (word$_i$ ∈ words-list; i=1 to total number of words within the word-list). Finally, *decision making phase* is applied on each word within the word-list to determine if it is a keyword or not.

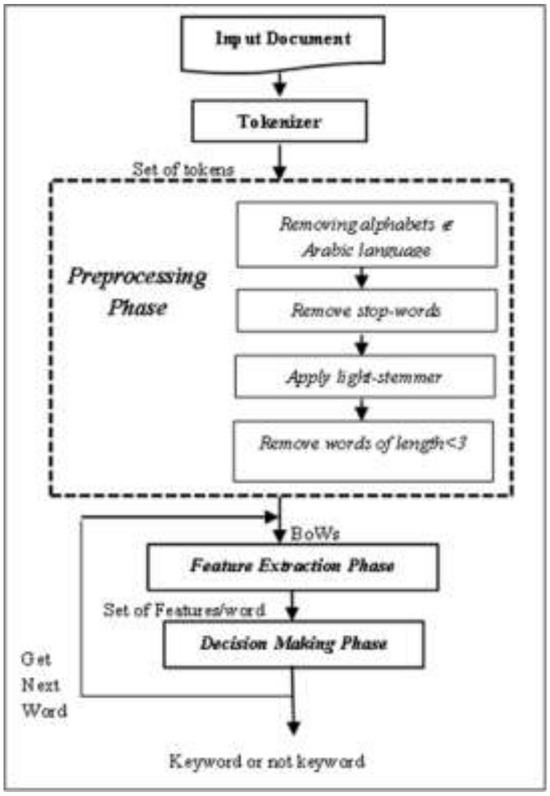

Figure 1. AAKE main model

A. Preprocessing Phase

As it is known, not all the *words* in the document have equal importance for representing the document semantics. Therefore, linguistic preprocessing steps are needed to specify the words that express the document's semantics, and generate the word-index. These words are nominated to be keywords [11]. To generate proper documents indices, in this work, non-Arabic letters, punctuation marks, digits, extra spaces, and the special symbols are removed. Arabic characters may have different forms. Therefore, it is important to make these Arabic characters have single canonical form (***Normalization***) [2]. El-Tanwin “ ً” is deleted; all forms of Alef “ “,“إ أ“,“ ” are replaced with “ا”; all Alif-maksura“ ى” are replaced with Ya“ي”; finally, all Ha' “ه” are replaced with Ta' marbota “ ة”. ***Stop-words are removed,*** as they are considered noise words [7]. These words play grammatical roles in the document and are not related to its content [11]. Stop words are the most common words in Arabic language. They appear in any text. Therefore, stop-words should be excluded from the word-list. Removing stop-words has the advantage of reducing the size of the candidate keywords. In this work, words ∈ stop-word list in [4] are removed. ***Light Stemming*** is applied to enhance keywords extraction process while retaining the words' meaning [4]. In this work, we applied the light stemming suggested in [18]. Finally, since most Arabic morphemes are defined by 3 characters or more. thus, words of length <3 letters are removed [18].

B. Features Extraction Phase

In AKE systems, keywords are specified based on the word features. The features could be statistical or semantic features. In this work, nine statistical features (illustrated in section (2.1)) are extracted for each word∈word-list. These features are TF, SF, T, FS, LS, NTF, RTF, NSF and RSF.

C. Decision Making Phase

In this phase, *unsupervised* clustering is used to specify the keywords of a document. Self-organized map (Kohonen neural network KNN) is used as unsupervised clustering technique to partition the word-list of the document into two classes, keywords and non-keywords classes. Figure (2) illustrates the KNN.

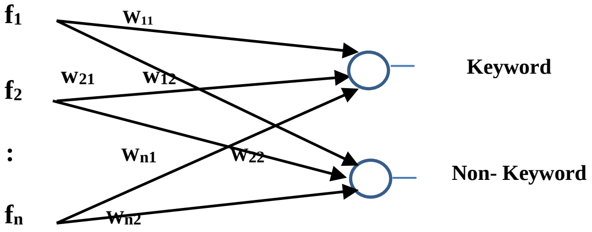

Figure 2. The architecture of the used KNN

The KNN consists of two layers, input layer and output layer. Number of nodes in the input layer equals number of word’s features. The output layer consists of two nodes indicating Keywords (if the word is judged as keyword, K is set to 1, otherwise, 0), and Non-keywords (if the word is judged as non-keyword, N is set to 1, otherwise, 0). In this work, we tested different feature-sets. The tested feature-sets ⊂ nine features. Therefore, number on input nodes $1 \leq n \leq 9$. Weights (W) are initialized randomly, the learning rate α=0.6.

IV. AAKE Evaluation

One of the problems researchers may face when evaluating Arabic natural-language systems is the lack of standard Arabic corpora. To evaluate AAKE performance, two datasets are used. The first one is 48 Arabic documents collected from Jordan Journal of Social Sciences (JJSS)), associated with their keywords list. The second dataset consisted of 24 document is collected by [9] from the Wikipedia. For Wikipedia documents, meta-tag is used a keywords. The text documents of the first dataset (JJSS) are considered short texts that consisted approximately from 3-5 paragraphs, while the text documents in the second dataset (Wikipedia) are considered as long text since each consisted from 10-15 paragraphs. The performance of AAKE is measured using average recall, precision, and F-measure. AAKE performance is compared with Sakhr keyword execrator system (as Sakhr is the only free Arabic keyword extractors available and it is considered a robust system). To specify the best statistical feature-set, that best suit Arabic keywords extraction, 12 different combination and subsets of 9 statistical features (TF, SF, T, FS, LS, NTF, RTF, NSF and RSF) are used. These sets are:

Set-1: < T, TF, SF, FS and LS >, all features without normalization or ration.

Set-2: < T, TF, FS and LS >, study the effect of removing SF on the performance.

Set-3: < TF, SF, FS and LS >, study the effect of removing T on the performance.

Set-4: < T, SF, FS and LS >, test the effect of removing TF on system performance.

Set-5: < T, TF, SF and LS >, study the effect of removing FS on the performance.

Set-6: < T, TF, SF and FS >, test the effect of removing LS on system performance.

Set- 7: test the ability of < TF > to extract keywords.

Set- 8: test the ability of < SF > to extract keywords.

Set-9: test the ability of keyword extraction of the two features < TF and T >.

Set-10: use NTF and NSF < T, NTF, NSF, FS and LS > to test the normalization effect.

Set-11: use RTF and RSF < T, RTF, RSF, FS and LS > to test the ratio effect on keyword extraction.

Set-12: use < T, NTF, RSF, FS and LS >, to test their effect on the system behavior.

From sets-1 contains 5 features. Set-2 to set-6 includes 4 features (each time, 1 feature is excluded to study its effect on the keyword extraction process). Feature set-7 to set-9 are selected based on the observation of the first 6 sets. Finally, set-10 to 12 are used to test normalized features (NTF, NSF), and ratio features (RTF, RSF). From these 12 different feature-sets, three goals will be achieved:

- The best feature set that improve the accuracy of keyword extraction is specified.
- The effect of normalizing TF and SF (i.e. NTF and NSF) will be illustrated.
- Illustrate the effect of using the ratios (RTF, RSF) instead of TF and SF on the system performance.

A. *Assessment of Experimental Results*

At the beginning, we will specify the best feature-set by comparing the average recall, precision, and F-measure of the 12 cases on the first dataset (JJSS). To specify the best feature set:

1) Extracted the keywords from each of the 48 documents.

2) Calculated the recall, precision, and F-measure for each document using Eq.s (6-8).

$$\mathrm{Recall} = \frac{|\mathrm{Keywords} \cap \mathrm{Rtrieved\,Words}|}{|\mathrm{Keywords}|} \tag{6}$$

$$\mathrm{Pr}\,ecision = \frac{|\mathrm{Keywords} \cap \mathrm{Rtrieved\,Words}|}{|\mathrm{Rtrieved\,Words}|} \tag{7}$$

$$F - measure = 2 \times \frac{\mathrm{Pr}\,ecision \times \mathrm{Re}\,call}{\mathrm{Pr}\,ecision + \mathrm{Re}\,call} \tag{8}$$

3) Find the average of recall, precision, and F-measure respectively.

Figure 3 shows AAKE performance (recall, precision, and F-measure) when applied on JJSS dataset. The best feature set is set-1 (T, TF, SF, FS, LS), the highest recall (52.63%), and comparable f-measure (44.35%) are achieved. Set-2 (T, TF, FS and LS) is the feature-set with highest precision (42.84%) and F-measure (44.72%). It is also important to point out that the TF is very important feature. The feature sets that does not include TF, RTF, or NTF shows the worst performance (as in sets 4, 8). The features that have least effect are T, SF, and LS (as with sets 2, 3 and 6). Finally, NTF and NSF showed better performance than RTF and RSF as in sets 10 and 11. From the result of the first experiment, it was found that the best feature sets are set-1 and set-2. Consequently, these two feature sets are tested with the second dataset (Wikipedia).

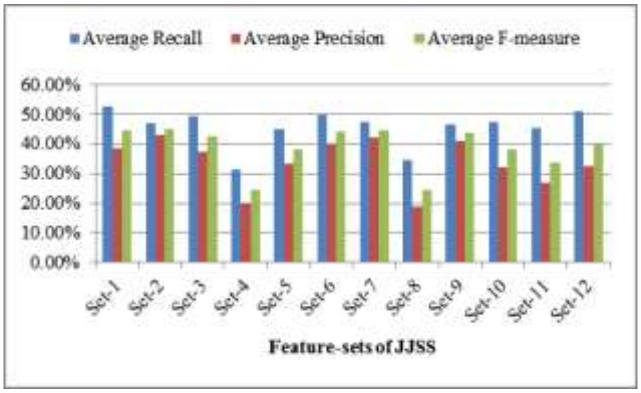

Figure 3. JJSS dataset: performance of AAKE

Figure 4 shows that feature set-2 gives better average recall (32%) and average f-measure (37.74%), while set-1 shows better average precision (51.90%). The overall performance of AAKE is better when applied on JJSS data set. This may be due to the nature of the Wikipedia dataset where the meta-tags are considered as keywords. But actually, not all the meta-tags are keywords. Meta-tags contains information concerning how to create the web page, how often it is updated, what the page is about, and which keywords represent the page's content.

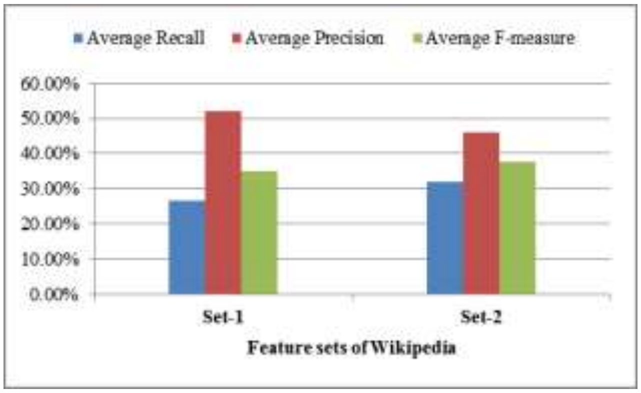

Figure 4. Wikipedia dataset: performance of AAKE

B. *Comparison between the Proposed System Results and Sakhr System*

To assure the performance of AAKE, it is important to compare AAKE with other AKE systems. In this work, we chose to compare the performance of AAKE and Sakhr system by applying them on the two datasets (JJSS and Wikipedia). Table (1) shows that AAKE has better recall than Sakhr with JJSS. This is because JJSS dataset does not contain a lot of names, celebration, places and dates, where the Sakhr system considers names, celebration, places and dates as keywords. With Wikipedia, Sakhr has better average recall. This due to fact that there are no actual keywords; meta-tags is used instead. Sakhr system considers names, celebration, places and dates as keywords, by comparing this type of keywords with the meta-tags, the recall and precision will be high. On the other hand, AAKE selects the keywords based on features statistics. By comparing the performance of AAKE with Sakhr system from precision and F-measure point of views, it is clearly seen that AAKE outperforms Sakhr system.

TABLE I. KEYWORD EXTRACTION PERFORMANCE OF SAKHR AND AAKE (F-M INDICATES THE F-MEASURE)

	Sakhr			**AAKE**		
	Recall	***Precision***	***F-M***	***Recall***	***Precision***	***F-M***
JJSS	34.25%	16.91%	22.64%	46.79%	42.84%	44.72%
Wikipedia	66%	9.80%	17.06%	32%	46%	37.74%

V. CONCLUSION

In this work, AKE from Arabic single text document approach based on statistical features is developed. The developed approach is a hybrid approach (linguistic, statistical, and machine learning). It was found that using Kohonen Neural network (as unsupervised machine learning approach) to cluster keywords based on simple linguistic and set of statistical features is a successful approach since, based on average F-measure, the system performance is comparable to other keyword extraction systems. F-measure is essential metrics since it represents a trade-off metrics between precision and recall. Therefore, it will be used to determine the best feature set. By comparing the performance of 12 combinations statistical features, it was found that set-2 is the best feature set. It was also found that TF is a significant feature, which highly affects the system performance. On the other hand, SF almost has no effect on the system performance, especially with short documents. By applying AAKE and Sakhr on JJSS and Wikipedia, using feature Set-2, it was found that AAKE outperforms Sakhr from F-measure point of view.

REFERENCES

[1] Gonenc Ercan, Ilyas Cicekli. (2007, Nov). “Using lexical chains for keyword extraction.” Information Processing and Management. 43(6), pp. 1705–1714

[2] Suhad A. Yousif, Venus W. Samawi, Islam Elkaban and Rached Zantout. (2015). “Enhancement of Arabic Text Classification Using Semantic Relations of Arabic WordNet.” Journal of Computer Science. [On-line].11(3), pp. 498-509. Available http://thescipub.com/PDF/jcssp.2015.498.509.pdf

[3] Zhang C., Wang H., Liu Y., WU D., Liao Y., Wang B. (2008, Sep.). “Automatic Keyword Extraction from Documents Using Conditional Random Fields.” Journal of Computational Information System. [On-line]. 4(3), pp 1169-1180. Available http://eprints.rclis.org/12305/

[4] Rehab Duwairi, Mohammad Nayef Al-Refai, Natheer Khasawneh. (2009, Jul.). “Feature Reduction Techniques for Arabic Text Categorization.” Journal of the American Society for Information Science and Technology. [On-line]. 60(11), pp.2347–2352. Available http://www.just.edu.jo/~rehab/j5.pdf

[5] El-Beltagy S., Rafea A. (2009, March). “KP-Miner: A Keyphrase Extraction System for English and Arabic Documents.” Information System. 34(1), pp. 132-144.

[6] Fatima T. AL-Khawaldeh, Venus W. Samawi. (2015). “Lexical Cohesion and Entailment based Segmentation for Arabic Text Summarization (LCEAS).” World of Computer Science and Information Technology Journal (WCSIT). [On-line]. 5(3), pp.51- 60. Available http://v1.wcsit.org/index.php/10-articles/187-lexical-cohesion-and-entailment-based-segmentation-for-arabic-text-summarization-lceas

[7] Nashat Mansour, Ramiz Haraty, Walid Daher, Manal Houri. (2008, July). “An auto-indexing method for Arabic text.” Information Processing and Management. 44(4), pp. 1538–1545.

[8] Khalid Al-Khatib, Amer Badarneh. (2010). “Automatic Extraction of Arabic Multi-Word Terms.” Proceedings of the International Multi-conference on Computer Science and Information Technology, IEEE. 5, 2010, pp. 411-418.

[9] Tarek El-shishtawy, Abdulwahab Al-sammak. (2012). “Arabic Keyphrase Extraction using Linguistic knowledge and Machine Learning Techniques.” Proceedings of the Second International Conference on Arabic Language Resources and Tools. 2012. Available http://arxiv.org/abs/1203.4605

[10] Arafat Awajan. (2015, March). "Keyword Extraction from Arabic Documents using Term Equivalence Classes." .14(2)

[11] Jo T. "Neural Based Approach to Keyword Extraction from Document." in Lecture notes in computer science: Computational Science and Its Applications — ICCSA 2003, vol. 2667. Vipin Kumar, Marina L. Gavrilova, Chih Jeng Kenneth Tan, Pierre L'Ecuyer, Springer Berlin Heidelberg, 2003, pp 456-461

[12] Kamal Sarkar, Mita Nasipuri and Suranjan Ghose. (2010, March). "A New Approach to Key-phrase Extraction Using Neural Networks." IJCSI International Journal of Computer Science Issues. [On-line]. 7(2), No 3, pp. 16-25. Available http://arxiv.org/ftp/arxiv/papers/1004/1004.3274.pdf

[13] Jasmeen Kaur,Vishal Gupta. (2010, Nov.). "Effective Approaches for Extraction of Keywords." IJCSI International Journal of Computer Science Issues. [On-line]. 7(6). pp144-148. Available http://ijcsi.org/papers/7-6-144-148.pdf

[14] Vishal Gupta, Gurpreet Singh Lehal (2011, Sep.). "Automatic Keywords Extraction for Punjabi Language." IJCSI International Journal of Computer Science Issues. [On-line]. 8(5), No 3, pp327-33. Available http://www.ijcsi.org/papers/IJCSI-8-5-3-327-331.pdf

[15] Chong L., Chen Y. (2009). "Text Summarization for Oil and Gas News Article." International Journal of Computer, Electrical, Automation, Control and Information Engineering. [On line]. 3(5), pp 1282-1285. Available http://waset.org/publications/4926/text-summarization-for-oil-and-gas-news-article

[16] Yutaka Matsuo, Mitsuru Ishizuka. (2004, March). "Keyword Extraction from a Single Document using Word Co-occurrence Statistical Information." International Journal on Artificial Intelligence Tools. 13(1), pp157-169.

[17] Sheng-Chai Chi, Chi-Chung Lee, and Tung-Chang Young. (2003). "A Three-layered Self-Organizing Map Neural Network for Clustering Analysis." Systemic Cybernetics and Informatics. [On line]. 1(6), pp 24-33. Available http://www.iiisci.org/Journal/CV$/sci/pdfs/P568803.pdf

[18] Motaz Saad and Wesam Ashour. "Arabic Morphological Tool for Text Mining," presented at the 6th International Conference on Electrical and Computer Systems (EECS'10), Lefke, North Cyprus, Nov 25-26, 2010. Available http://site.iugaza.edu.ps/wp-content/uploads/mksaad-ArabicMorphologicalToolsforTextMining-EECS10-rev9.pdf

.