Extractive Summarization Method for Arabic Text - ESMAT

Mohammed Salem Binwahlan

Department of Computer Science, Faculty of Applied Science,
Hadhramout University, Yemen

Abstract— **Due to the huge and rapid growth of online data makes search such massive data collections and finding the relevant information a tough task and time consumption. For this reason, research on automatic summarization techniques has received much attention from industry and academia. Unlike English text which has received much attention of the researchers in this field, Arabic text is still lake to such serious investigations. This reason gave the author of this paper, strong motivation to participate in a pushing Arabic language into the concern domain of automatic text summarization researchers by proposing an extractive summarization method. The proposed method generates a summary of an original document based on a linear combination of text features having different structures. Five summarizers (AQBTSS, Gen–Summ, LSA–Summ, Sakhr and Baseline–1) are used in this study as benchmarks. The proposed method and the benchmarks are evaluated using EASC – the Essex Arabic Summaries Corpus. The results showed that the proposed method performs well, based on recall, precision and average scores, more than the five benchmarks. A good performance achieved by the proposed method proved that the focus on those more complicated features, rather than simple ones, could guide to the most important content of any document.**

Keywords— **Automatic text summarization, summary, sentence similarity, term frequency, text feature.**

I. INTRODUCTION

Due to the huge and rapid growth of online data makes search such massive data collections and finding the relevant information a tough task and time consumption. For this reason, research on automatic summarization techniques has received much attention from industry and academia. Automatic text summarization is the process of scanning a full text for discovering its parts bearing the most important meaning, and presenting those parts in a limited size space. The requirement of including the most informative parts in that limited size space (which is called a summary of the full text) addresses a big challenge. Such challenge forces the researchers in the area of text summarization to deal with it in two directions, the first one is how to determine the most important parts of the full text and second one is how to control the inclusion of those parts in the limited size space (the summary). The purpose of text summarization is to provide a good indication (the summary) to the full text content, which helps readers to make a decision to read the full text of document or not. Reading the summary instead of the full text can save the time and effort. To this end, many text summarization algorithms have been proposed based on different techniques and different methodologies. Those proposed algorithms were classified into two main categories, extractive and abstractive [1]. Extractive algorithms insert the most important parts of the original document, without changing the structure of those parts (simple copy), into the final summary. Similar to extractive algorithms, abstractive algorithms insert the most important parts of the original document into the final summary, but after editing the structure of those parts (perform paraphrasing). And this makes the abstractive algorithms more complicated than extractive algorithms.

The cornerstone of automatic text summarization systems is those approaches which dates back to the 1950s and 1960s [2; 3]. Such approaches depend on a linear combination of shallow features of text units to calculate the score of these units [2; 3; 4]. Luhn [2] proposed that the word significance is determined by frequency of its occurrence and the significance of sentence is determined by the relative position of its words. A combination of these two measurements determines the significance factor of a sentence. The highest score sentences are chosen as summary sentences "auto-abstract" where as the sentences are reordered based on their significance order. Edmundson [3] presented summarization system to generate extracts in which four features are used: word frequency, positional importance, cue words, and title or heading words. Each sentence is scored by the weights of the four features. Each feature is given a weight manually. The advantages of these approaches are simplicity and efficiency. In Baxendale's study [4], a sentence is selected as a candidate for the summary based on its position. The sentence appearing in the beginning and the end of the paragraph has been given more significance. Zechner [5] presented a pure statistical abstract-based system employing only tf*idf weight to score the text sentences. The system is a neutral of domain knowledge and text characteristics.

The features are the main entries in text summarization [2; 3], each feature plays a different role for showing the most important content. Extraction of the most important content is affected by the features selected. So, the feature selection received much attention by many approaches [2; 3; 4; 5; 6]. Therefore, this work focuses on those more complicated features rather than simple features.

Unlike English text which has received much attention of the researchers in this field, Arabic text is still lake to such serious investigations. This reason gave the author of this paper, strong motivation to participate in a pushing Arabic language into the concern domain of automatic text summarization researchers by proposing an extractive summarization method based on a number of features of different structures. The proposed method is similar to [2; 3; 5], in terms of scoring a sentence based on a linear combination of text features, and differs from those techniques by focusing on more complicated features; because such features can guide to the most important content of any document [6]. Most features used in the proposed method are to identify the relation between each sentence and the document containing it. Many works on text summarization considered concepts (or key phrase) as key entry to discovery most important text units in a document [7; 8; 9; 10]. The proposed method is similar to these methods in terms of considering the importance of multi-word concepts, but it is more similar to [10]. The proposed method creates the multi-word concepts as bi-grams (two words), it divides the sentence into a number of bi-grams (each bi-gram shares its first word with the its precedent bi-gram (if any) and shares its second word with the its subsequent bi-gram (if any)). HIRAO et. al. [10] divides the sentence into a number of uni-grams, bi-grams, tri-grams, and there is no shared words between any two successive divisions. Another complicated feature used by the proposed method, and consider more important, is log entropy which was used by Gulcin et. al. [11] for Turkish Texts. Another feature that can reflect the relation between each sentence and the document containing it is a sentence similarity to its document [12; 13].

II. RELATED WORK

Recently, Arabic language gained an attention of automatic text summarization researchers. In this section, a number of works, which have been done on Arabic text, are reviewed in general. Deep investigation will be for systems which were evaluated using EASC – the Essex Arabic Summaries Corpus [14]; because the current study is evaluated using the same corpus as well.

The first attempt to push Arabic language into the concern domain of automatic text summarization researchers was in 2004 when Douzidia and Lapalme [15] proposed an extractive approach based summarizer, called “Lakhas, that generates very short (headline) summary. The proposed summarizer, first reads Arabic document, builds an abstract representation of the whole document, and then selects a highly relevant sentences to form a summary of that document. The summary length was limited to ten words. The scoring mechanism considers four features: sentence position in the document, number of subject terms (i.e., words that appear in the headline) in the sentence, number of “indicative words” in the document, and the tf.idf value of each word in the sentence. “Lakhas” participated in the DUC-2004 task 3 of generating very short summaries (75-bytes).

Conroy et. al. [16] introduced a summarization system called CLASSY, in which, to score a sentence, oracle score is used, which enable to determine the number of terms shared between abstract and a sentence. That score led to propose a new score called “approximate oracle”, the new score determines whether a sentence is include in the summary or not. A new Traveling Salesperson (TSP) formulation was used to order sentences in the summary. CLASSY achieved a high performance in the DUC 2006 competition.

El-Haj et. al. [17] presented two Arabic summarization systems based on vector space model (VSM) [18]: Arabic query-based text summarization system (AQBTSS) and Arabic concept-based text summarization system (ACBTSS). Both systems perform the summarization process through two steps. In the first step, the users should search the document collection to retrieve high relevant documents to a given query (a specific user’s query or a set of words representing some concepts). The users then choose the most relevant document and pass it to the second step. In the second step, the two systems behave differently. AQBTSS matches each sentence in the selected document with the given query and retrieves the most relevant sentences as a summary. Whereas ACBTSS replaces the given query with a set of words representing some concept. In both system, term frequency (TF) and inverse document frequency (IDF) are used as weighting scheme to determine the relevance degree which is represented in the VSM. The problem with those two systems is that they require the user to participate in the first step of the summarization process. Elhaj [19] proposed two generic extractive single-document summarisers for Arabic, based on the idea of [17], called “Gen–Summ” and “LSA–Summ”. In these two systems, the user participation in the summarization process, as in [17], was removed. Gen–Summ is special query-based summarizer; because it uses the document’s first sentence as a query. In Gen–Summ, VSM [18] is used as in [17]. The difference between LSA–Summ and Gen–Summ is that LSA–Summ used latent semantic analysis (LSA) [20] instead of the VSM. The two systems generate a summary of no more than 50% of the document’s words count. AQBTSS, Gen–Summ, LSA–Summ, Sakhr (a commercial online Arabic text summariser available on the web1) and Baseline–1 (first sentence of a document was taken as a summary) were evaluated in [19] using EASC – the Essex Arabic Summaries Corpus [14]. These five summarizers (AQBTSS, Gen–Summ, LSA–Summ, Sakhr and Baseline–1) are used in this study as benchmarks; because the proposed method will be evaluated using the same dataset (EASC – the Essex Arabic Summaries Corpus [14]

Ibrahim et. al.[7] presented an Arabic text query based, single document summarizer using knowledge base. The proposed summarizer consists of two modules. The first module is for building the knowledge, in which the multi-word concepts (words that frequently appear together more often than can be expected by chance) are extracted. A linguistic and statistical knowledge based method called " C-/NC-value " [21] was used to discover the multi-word concepts. In the first module also, the relations of “is-a” and “has-a” between concepts are discovered. The second module is for summarization, it starts with a user's query expansion, where double expansion process is run for the given query. Firstly, the query is expanded using Arabic WordNet database and secondly it is expanded using the knowledge base of concepts and relations. Then the second module is finalized by the summarization of a document, where the words of the document are matched, first with the original query and second with the expanded query. Each sentence receives score of 1 or 0.5 for each exact matching between each of its words with the original and the expanded query respectively. Those sentences whose high scores summation form the final summary. for evaluation of the proposed summarizer, Essex Arabic Summaries Corpus[14] was used. The current study related to [7] study in terms of the multi-word concepts (with difference of creation and usage) and the evaluation Corpus.

Ibrahim et. al. [8] modified their work in [7], they trained the decision tree algorithm (C4.5) on a set of features extracted from the original documents. The summary is generated by including those sentences selected by the C4.5 trained model. For the testing purpose, the Essex Arabic Summaries Corpus (EASC) [13] was used.

Sobh et. al. [22] presented a trainable Arabic extractive generic text summarizer based on the Bayesian classifier. For each sentence in a document a number of features are extracted to form a feature vector for that sentence. The feature vector and a training corpus are used by the Bayesian classifier to classify each sentence to be in summary or out of summary.

Sobh et. al. [23] presented an Arabic trainable extractive text summarization system based on Bayesian and Genetic Programming (GP) classifiers. Both classifiers mark each document sentence as summary sentence or out of summary. The results of the two classifiers are merged, first using union operation and second using intersection operation. Four types of summary are generated: Bayesian based summary, GP based summary, Bayesian and GP based summary (intersection), and Bayesian or GP based summary (union)

Ibrahim et. al.[24] proposed Arabic text summarizer based on rhetorical structure theory (RST) and vector space model (VSM). The most important paragraphs in the document is identified using RST based on functional and semantic criteria. Then, those most important paragraphs are represented, and ordered based on their cosine similarity score, in VSM.

Azmi and Al-Thanyyan [25] presented an Arabic summarizer consisting of two modules. The first module uses rhetorical structure theory (RST) to extract a set of sentences which is used as initial summary. in the second module, the initial summary sentences are scored based on summing up the weights of a number of features, then the sentences having highest scores are selected to form the final summary.

Alotaiby et. al. [26] introduced two Arabic automatic headline generation methods. The first is an extractive method based on character cross-correlation, and the second one is an abstractive method based on the hidden Markov model (HMM).

El-Haj and Rayson [27] proposed a single-document and multi-document summarizers, in which a summary is generated by selecting those sentences with high scores. The documents sentences are scored in three steps. Firstly, word frequency lists from the corpus are produced. Secondly, the log likelihood score for each word in the word frequency lists is calculated. Thirdly, the sentence score is calculated by summing up the log likelihood scores of its words.

III. METHOD

The proposed method generates a summary of an original document based on a linear combination of text features; because such features can guide to the most important content of any document [6]. Most features used in the proposed method are to identify the relation between each sentence and the document containing it. The summarization process is done by the proposed method through two steps: pre-processing and feature extraction, and summary generation.

A. Pre-processing and feature extraction

The proposed method makes use of six text features (average TF-ISF(ATI), sentence length(SL), sentence position(SP), sentence similarity to document (SSD), sentence concepts(SC), and log entropy(LE)). A preprocessing of the original document, like breaking the input document into a list of sentences, stemming and removing stop words, is done first, and then those features are extract. The features are as follows:

1) Sentence concepts(SC) feature: A level-1 The extraction of this feature is similar to [10]. The proposed method creates the multi-word concepts as bi-grams (two words), it divides the sentence into a number of bi-grams (each bi-gram shares its first word with the its precedent bi-gram (if any) and shares its second word with the its subsequent bi-gram (if any)). Whereas HIRAO et. al.[10] divides the sentence into a number of uni-grams, bi-grams, tri-grams, and there is no shared words between any two successive divisions.

After extraction of each sentences concepts, every sentence is scored based on the following equation:

$$SC(s_j) = \left[\left[\sum_{i=1}^{nc(s_j)} \sum_{k=1}^{m} \sum_{q=1}^{nc(s_k)} c_i \times c_q \right] \Big/ MAX\_SC \right] \mid j \neq k \qquad Eq.1$$

Where *SC* is a sentence concept score, *nc* is a number of concepts in a sentence, *m* is a number of sentences in the document and *MAX_SC* is a maximum

2) Log entropy(LE) feature: Another complicated feature used by the proposed method, and consider more important , is log entropy which was used by Gulcin et. al. [11] for Turkish Texts.

$$sum = \sum_{i} p(i,j)\log_2(p(i,j)) \qquad Eq.2$$

$$global(i) = 1 + \frac{sum}{\log_2(m)} \qquad Eq.3$$

$$local(i,j) = \log_2(1 + f(i,j)) \qquad Eq.4$$

$$EL(s_j) = global \times local \qquad Eq.5$$

where *p(i,j)* is the probability of word *i* that is appeared in sentence *j*, *f(i,j)* is the number of times word *i* appeared in sentence *j*, and *m* is the number of sentences in the document.

3) *Average TF-ISF(ATI) feature:* One of the key entries to find the important sentences in an document is the term frequency [5]. The proposed method evaluates each document sentence using average Tf-Isf weights summation of its words. Tf.Idf method [28] is modified to calculate Tf-Isf weight Eq.6 , where the document parameter is replaced by the sentence parameter and document collection parameter is replaced by the document parameter:

$$TI \; t_i \; = Tf\_Isf \; t_i \; = tf \; t_i \; + \; \log TNT(D)/TON(t_i, D) \qquad Eq.6$$

4) where $tf(t_i)$ is the term frequency of i^{th} word in the sentence, *TNT(D)* is total number of terms in the document, and $TON(t_i, D)$ is total number of t_i occurrences in the document.

$$ATI(s) = \left(\frac{\sum_{i=1}^{n} TI(t_i, s)}{TNT(D)} \right) \qquad Eq.7$$

Where *ATI(s)* is average *Tf-Isf* score of sentence *s*, *n* is a number terms in sentence *s*, and $TI(t_i,s)$ is *Tf-Isf* of term *i* in sentence *s*.

5) *Sentence similarity to document (SSD) feature:* Another feature, that can reflect the relation between each sentence and the document containing it, is a sentence similarity to its document [11; 12].

$$SSD(s_j) = \sum_{k=1, k \neq j}^{m} sim(s_j, s_k) \qquad Eq.8$$

Where *m* is a number of sentences in the document (*D*)

To calculate the sentence similarity between two sentences s_j and s_k, cosine similarity measure as in *Eq.9* is used[29]:

$$sim(s_j, s_k) = \frac{\sum_{i=1}^{n} (w_i, s_j) \bullet (w_i, s_k)}{\sqrt{\sum_{i=1}^{n} (w_i, s_j)^2} \cdot \sqrt{\sum_{i=1}^{n} (w_i, s_k)^2}} \qquad Eq.9$$

Where w_i is *Tf-Isf (TI)* of term t_i in the sentence s_i or s_j, *n* is a number terms in sentence *s*.

6) *Sentence length(SL) feature:* The longer sentences in the document seem to carry more important meaning. The length of a sentence is the total number of its words [30].

$$SL \; s_i \; = TNT(s_j) / \max\_SL \qquad Eq.10$$

Where $TNT(S_j)$ is a total number of words in s_j and *max_SL* is the max sentence length in the document.

7) *Sentence position(SP) feature:* A sentence that sentences appear in the beginning of a paragraph in the original document has high importance [1; 31; 32; 33]. The proposed method give score of 1 to each paragraph starting sentence and score of 0 to the other sentences in the paragraph.

$$SP(s_j) = \begin{cases} 1 & if \; paragraph \; starting \; sentence \\ 0 & otherwise \end{cases} \qquad Eq.11$$

B. *Summary Generation*

To generate a summary of the input document by the proposed method, the document sentences are score using *Eq.12* firstly, the scored sentences are ranked based their scores secondly, and finally, those sentences having highest scores are picked up as a summary sentences.

$$Sent\_Score \; S_j \; = Avg \; ATI(s_j) + SL(s_j) + SP(s_i) + SSD(s_j) + SC(s_j) + LE(s_j) \qquad Eq.12$$

IV. EXPERIMENTAL DESIGN

The proposed method produces a summary of an original document through exploiting a linear combination of six text features; because such features can guide to the most important content of any document [6]. For the evaluation of the proposed method, The EASC [13] is used, it is an Arabic natural language resources. It contains 153 Arabic articles cover different topics (art & music, education, environment, finance, health, politics, religion, science & technology, sports, and tourism) and 765 human-generated extractive summaries of those articles, five human summaries for each original document. These summaries were generated using Mechanical Turk [13]. This study follows the same strategy explained in [18] for reproducing three types of human summaries (referred as level 3, level 2 and level all) based on the five human summaries of each original document. The summary of the first type (Level 3) contains all sentences appeared in at least three of the five human summaries. The summary of the second type (Level 2) contains all sentences appeared in at least two of the five human summaries. Finally, The summary of the third type (Level all) contains all sentences appeared in at least one of the five human summaries.

Summaries of length 50% is created by the proposed method. To evaluate those summaries, ROUGE (Recall-Oriented Understudy for Gisting Evaluation) toolkit [34] is used. ROUGE compares a system generated summary against a human generated summary to measure the quality of the system summary. ROUGE is the main metric in the DUC text summarization evaluations. It has different variants. In the experiment of this study, ROUGE-N (N = 2) is used. The reason for selecting this measure is that measures works well for single document summarization [34]. One of ROUGE settings is to determine a number of words to be selected from a summary being evaluated, in this study, that setting called "ROUGE-cut", where each summary is evaluated based on

ROUGE-cut 100. The reason behind the determination of summaries lengths as 50%, is what reported in [18], that the length of human summary is not same for three human summary types (level all, level 2 and level 3), it falls in a range between five words and 515 words with an average of 114 words (five sentences) per summary. the average of words per the summary of, the first type (level all), the second type (level 2), the third type (level 3) is 250 words, 175 words, 98 words respectively. The summary length was also limited in text analysis conference (TAC) to be in a range from 240 to 250 words.

Five summarizers (AQBTSS, Gen–Summ, LSA–Summ, Sakhr[1] and Baseline–1) have been evaluated in [18] using the same dataset (EASC – the Essex Arabic Summaries Corpus [13]. This study is also evaluated using the same dataset, therefore, those five summarizers are used as benchmarks, to compare their performances with the performance of the proposed method.

V. EXPERIMENTAL RESULTS AND DISCUSSION

The proposed method and the benchmarks are used to create a summary for each document in the document set used in this study. Each system created a good summary compared to the reference (human) summary. The results, drawn in Table 1, were evaluated using the first 100 words from the beginning of each summary (ROUGE-cut 100). The summary length was considered as 50% of the original document length. The performance of the proposed method is compared with the five summarizers (AQBTSS, Gen–Summ, LSA–Summ, Sakhr[1] and Baseline–1), which are used in this study as benchmarks based on ROUGE-2 measure only, because those benchmarks have been evaluated in [18] using that measure only. Based on average of ROUGE-2 measure scores, as shown in Figure 1**,** it is realized that the proposed method defeats the all five summarizers (AQBTSS, Gen–Summ, LSA–Summ, Sakhr[1] and Baseline–1).

By looking at the evaluation results of (shown in Table 1), the proposed method and the five benchmarks (where the summaries created by the proposed method and benchmarks were compared with the human summary of first type "level all"), a big difference can be noticed between the performance of, the proposed method and the five benchmarks, where the proposed method performs well, based on recall, precision and average scores, more than the five benchmarks.

TABLE I

COMPARISON OF THE PROPOSED METHOD (ESMAT) (SUMMARY LENGTH 50%, ROUGE-CUT: 100) WITH SAKHR, AQBTSS, LSA–SUMM, GEN–SUMM AND BASELINE–1 BASED ON RECALL, PRECISION AND AVERAGE OF, ROUGE -2 MEASURE OF THE THREE LEVELS (LEVEL ALL, LEVEL 2, AND LEVEL 3). (NO STEMMER)

Level	**Measure**		**ESMAT 100/50**	**Sakhr**	**AQBTSS**	**LSA–Summ**	**Gen–Summ**	**Baseline–1**
all	**ROUGE-2**	R	68.863	44.62	35.23	51.81	50.42	16.77
		P	69.622	42.09	43.70	39.35	40.12	66.41
		Avg	**69.184**	43.32	39.01	44.73	44.68	26.78
2	**ROUGE-2**	R	65.555	43.39	34.96	51.48	49.11	16.06
		P	65.234	64.87	66.44	61.66	60.88	99.35
		Avg	**65.269**	52.00	45.81	56.11	54.36	27.64
3	**ROUGE-2**	R	58.999	53.52	44.51	60.49	59.85	28.96
		P	48.841	43.32	49.27	41.71	42.36	88.24
		Avg	**51.862**	47.88	46.77	49.38	49.61	43.61

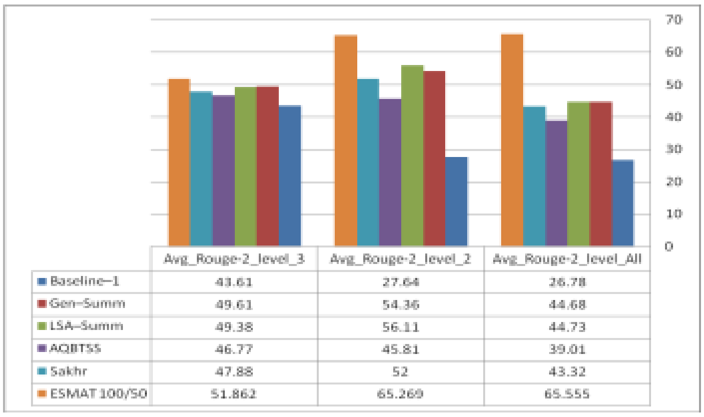

Figure 1: Comparison of the proposed method (ESMAT) with Sakhr, AQBTSS, LSA–Summ, Gen–Summ and Baseline–1 based on average ROUGE -2 of the three levels (Level all, Level 2, and Level 3).

By observing the results evaluation using the human summary of second type "level 2"), it can be seen that the results of the proposed method are better on average than the results of the five benchmarks. Some benchmarks recorded precision scores higher than the proposed method recorded, especially, Baseline–1 benchmark. This gives indication that the summary created by the proposed method contains many relevant sentences more than the summaries of the five benchmarks contain, which is reflected by recall scores, and its length more longer than the human summary of second type "level 2", which is reflected by precision scores. Explanation of such finding is that the human summary length is not bound to specific percentage of the original document. The same thing can be said on the results of, the proposed method and the five benchmarks (where the summaries created by the proposed method and benchmarks were compared with the human summary of third type "level 3"), it can be realized that the most results of the proposed method are better on recall scores than the results of the five benchmarks. Whereas the precision scores of two benchmarks (AQBTSS and Baseline–1) are better than the precision scores of the proposed method. On average, the proposed method result (with ROUGE cut 100) recorded score better than others.

REFERENCES

[1] Mani, I. (2001). Automatic Summarization. (1st ed.). Amsterdam: John Benjamins Publishing Company.

[2] Luhn, H. P. (1958). The Automatic Creation of Literature Abstracts. IBM Journal of Research and Development. 2(92), 159-165.

[3] Edmundson, H. P. (1969). New Methods in Automatic Extracting. Journal of the Association for Computing Machinery. 16(2), 264-285.

[4] Baxendale, P. (1958). Machine-made Index for Technical Literature - an Experiment. IBM Journal of Research Development. 2(4), 354-361.

[5] Zechner, K. (1996). Fast Generation of Abstracts from General Domain Text Corpora by Extracting Relevant Sentences. In Proceedings of the 16th International Conference on Computational Linguistics. 986–989, Copenhagen, Denmark.

[6] Binwahlan, M. S., Salim, N., & Suanmali, L. (2009b). Swarm based features selection for text summarization. IJCSNS International Journal of Computer Science and Network Security, 9(1), 175–179.

[7] Ibrahim Imam, Nihal Nounou, Alaa Hamouda, Hebat Allah Abdul Khalek. Query Based Arabic Text Summarization. International Journal of Computer Science And Technology. 4(2), 2013, Pp. 35-39

[8] Imam I. , Hamouda A., Abdul Khalek H, A. An Ontology-based Summarization System for Arabic Documents. International Journal of Computer Applications Volume 74– No.17, 2013, pp.0975 – 8887

[9] D'Avanzo E., Magnini B., Valli A. Keyphrase Extraction for Summarization Purposes: The LAKE System at DUC2004. In Proceedings of the 4th Document Understanding Conferences. DUC.

[10] HIRAO T., SUZUKI J., ISOZAKI H. and MAEDA E.. NTT's Multiple Document Summarization System for DUC 2004. In Proceedings of the 4th Document Understanding Conferences. DUC.

[11] Gulcin. M., Ilyas O., Cicekli F., Alpaslan N. Text Summarization of Turkish Texts using Latent Semantic Analysis. Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010), pages 869–876, Beijing, August 2010

[12] Ko Y., Seo J. An effective sentence-extraction technique using contextual information and statistical approaches for text summarization. Pattern Recognition Letters 29 (2008) 1366–1371

[13] Kim, J., Kim, J., Hwang, D., 2001. Korean text summarization using an Aggregation Similarity. In: Proc. 5th Internat. Workshop Information Retrieval with Asian Languages, pp. 111–118.

[14] Summarisation Corpora, [Online] Available: http//:privatewww.essex.ac.uk/~melhaj/easc.htm, (14-01-2013.)

[15] F. Douzidia and G. Lapalme. 2004. Lakhas, an Arabic Summarising System. In Proceedings of the 4th Document Understanding Conferences , pages 128–135. DUC.

[16] J. Conroy, J. Schlesinger, D. O'Leary, and J. Goldstein. Back to Basics: CLASSY 2006. In Proceedings of the 6th Document Understanding Conferences. DUC.

[17] El-Haj et. al.] M. El-Haj, U. Kruschwitz, and C. Fox. Experimenting with Automatic Text Summarization for Arabic. In Zygmunt Vetulani, editor, 4th Language and Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics, LTC'09, "Lecture Notes in Artificial Intelligence", pages 490–499, Poznan, Poland, 2009. Springer.

[18] Salton, G., Wong A., and Yang, S. 1975. A Vector Space Model for Automatic Indexing. Communications of the ACM, vol. 18, no. 11, (pp. 613–620).

[19] Elhaj M. Multi-document Arabic Text Summarisation. PhD thesis, 2012, University of Essex

[20] Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman. Indexing by Latent Semantic Analysis. Journal of the American Society for Information Science, 41(6):391–407, 1990.

[21] Sophia Ananiadou, Hideki Mima,"An Application and Evaluation of the C/NC-value Approach for the Automatic term Recognition of Multi-Word units in Japanese," International Journal of Terminology, Vol. 6, No. 2, pp. 175–194, 2000.

[22] Ibrahim Sobh, Nevin Darwish, Magda Fayek. A Trainable Arabic Bayesian Extractive Generic Text Summarizer.

[23] Ibrahim Sobh, Nevine Darwish , Magda Fayek. Evaluation Approaches for an Arabic Extractive Generic Text Summarization System.

[24] Ahmed Ibrahim, Tarek Elghazaly, Mervat Gheith. A Novel Arabic Text Summarization Model Based on Rhetorical Structure Theory and Vector Space Model. International Journal of Computational Linguistics and Natural Language Processing. Vol 2 Issue 8 August 2013

[25] Aqil M. Azmi, Suha Al-Thanyyan. Computer Speech and Language 26 (2012) 260–273 A text summarizer for Arabic _

[26] Fahad Alotaiby, Salah Foda and Ibrahim Alkharashi. New approaches to automatic headline generation for Arabic

documents. Journal of Engineering and Computer Innovations Vol. 3(1), pp. 11-25, February 2012

[27] Mahmoud El-Haj and Paul Rayson. Using a Keyness Metric for Single and Multi Document Summarisation Proceedings of the MultiLing 2013 Workshop on Multilingual Multi-document Summarization, pages 64–71, Sofia, Bulgaria, August 9 2013. C 2013 Association for Computational Linguistics

[28] Salton, G., 1989. Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer. Addison-Wesley Publishing Company.

[29] Horacio Saggion and Robert Gaizauskas. Multi-document summarization by cluster/profile relevance and redundancy removal. In Proceedings of the 4th Document Understanding Conferences. DUC.

[30] Chikashi Nobata and Satoshi Sekine.CRL/NYU Summarization System at DUC-2004. In Proceedings of the 4th Document Understanding Conferences. DUC.

[31] Lin, C. Y. and Hovy, E. (1997). Identifying Topics by Position. In Proceedings of the Fifth conference on applied natural language processing. March. San Francisco, CA, USA, 283-290.

[32] Hovy, E. H. and C-Y. Lin. (1999). Automated Text Summarization in SUMMARIST. In Mani I. and Maybury M. (Eds.). *Advances in Automated Text Summarization*. (pp. 81–94). Cambridge: MIT Press.

[33] Enrique Alfonseca, Jos´e Mar´ıa Guirao, Antonio Moreno-Sandoval. Description of the UAM system for generating very short summaries at DUC-2004. In Proceedings of the 4th Document Understanding Conferences. DUC.

[34] Lin, C. Y. (2004). Rouge: A Package for Automatic Evaluation of Summaries. *Proceedings of the Workshop on Text Summarization Branches Out, 42nd Annual Meeting of the Association for Computational Linguistics*. 25–26 July. Barcelona, Spain, 74-81.