Minimum Redundancy and Maximum Relevance for Single and multi-document Arabic Text Summarization

Houda Oufaida, Omar Nouali, Philippe Blache

Journal of King Saud University – Computer and Information Sciences (2014) **26**, 450–461

King Saud University

Journal of King Saud University – Computer and Information Sciences

www.ksu.edu.sa
www.sciencedirect.com

Minimum redundancy and maximum relevance for single and multi-document Arabic text summarization

Houda Oufaida [a,*], **Omar Nouali** [b], **Philippe Blache** [c]

[a] *Ecole Nationale Supérieure d'Informatique (ESI), Algiers, Algeria*
[b] *Research Center on Scientific and Technical Information (CERIST), Algiers, Algeria*
[c] *Aix Marseille Université, CNRS, LPL UMR 7309, 13604 Aix en Provence, France*

Available online 28 September 2014

KEYWORDS

Arabic text summarization;
Sentence extraction;
mRMR;
Minimum redundancy;
Maximum relevance

Abstract Automatic text summarization aims to produce summaries for one or more texts using machine techniques. In this paper, we propose a novel statistical summarization system for Arabic texts. Our system uses a clustering algorithm and an adapted discriminant analysis method: mRMR (minimum redundancy and maximum relevance) to score terms. Through mRMR analysis, terms are ranked according to their discriminant and coverage power. Second, we propose a novel sentence extraction algorithm which selects sentences with top ranked terms and maximum diversity. Our system uses minimal language-dependant processing: sentence splitting, tokenization and root extraction. Experimental results on EASC and TAC 2011 MultiLingual datasets showed that our proposed approach is competitive to the state of the art systems.

1. Introduction

Automatic summarization has received considerable attention in the past several years. Because it is a relatively old field (Luhn, 1958; Edmundson, 1969), the rapid growth of available documents in digital format was like "a breath of fresh air" to the field and has highlighted the importance of developing specific tools to find relevant information. Arabic documents are no exception. Indeed, Arabic content on the Internet has undergone a constant expansion: Arabic websites were ranked eighth at 3% [1] in April 2013, and there were more than 255 thousand Arabic Wikipedia articles in December 2013. Moreover, Arabic is the fifth most spoken language in the world,[2] and Arabic Internet users were ranked seventh at 3% in May 2011.

Recently, new tasks and challenges arose such as multi-document, multilingual and guided and updated summaries and gave a new boost to the automatic summarization field. Cross-lingual and multilingual summarizations received

* Corresponding author.
E-mail addresses: h_oufaida@esi.dz (H. Oufaida), onouali@mail.cerist.dz (O. Nouali), blache@lpl-aix.fr (P. Blache).

Peer review under responsibility of King Saud University.

ELSEVIER Production and hosting by Elsevier

[1] http://en.wikipedia.org/w/index.php?title=Languages_used_on_the_Internet&oldid=581612018.
[2] http://www.ethnologue.com/statistics/size.

http://dx.doi.org/10.1016/j.jksuci.2014.06.008

considerable attention and made some interesting multilingual datasets available.

The goal of text summarization is to produce a condensed version of one or more texts using computer techniques. This will help the reader to decide if a document contains needed information with minimum effort and time loss. Thus, a good summarizer should find key information and omit redundant information. For example, if we perform a search on a particular subject, such as "Arabic text summarization", we will find a multitude of documents. Among them, some are very interesting, and others are less relevant. Sorting these documents is a tedious task, which will take significant time. Thus, tools such as single- and multi-document summarization systems are quite useful.

Early summarization systems used natural language processing (NLP)-based techniques in a bid to *understand* the source text and *generate* new sentences to form an *abstract*: paraphrasing identification and information fusion, for example Barzilay and McKeown (2005). In a few other cases, NLP techniques were used to identify salient sentences, such as the use of rhetorical analysis RST (Marcu, 1998). However, these techniques are not yet mature; they still require heavy NLP processing often based on limited and language-dependent resources. Moreover, considerable work remains for Arabic NLP to reach the actual level of English NLP tools, for example: sentence splitting, tokenization, part-of-speech tagging, named entity recognition, and anaphora resolution. These basic NLP tasks have relatively acceptable performance for English and were used in many state of the art summarization systems. However, Arabic NLP systems are still in an early stage. Thus, developing effective Arabic summarization systems based on heavy NLP techniques is not yet possible.

Recently, statistical techniques have proved their performance and gained more ground. In this paper, we propose an Arabic statistical summarization method, which uses light language-dependent information. Our method extracts relevant sentences from single and multiple Arabic documents by maintaining minimum redundancy and maximum relevance. To achieve this, we first proceed to document preprocessing: sentence splitting, tokenization, stop words removal and root extraction. Second, we build a [Sentences $\times$ Terms] matrix, where each entry corresponds to the term's weight in the sentence. Third, we build a sentence-to-sentence similarity/distance matrix and perform clustering to put similar sentences in the same cluster. Fourth, we apply an adapted discriminant analysis method: minimum redundancy maximum relevance (mRMR) (Peng et al., 2005) to select most relevant terms from input Arabic document/documents with minimum redundancy. Then, a score is assigned to each sentence based on the new mRMR weights of its terms. Finally, n sentences are selected to construct the output summary, n depending on the required summary size.

This paper is organized as follows: we first introduce a brief review of related work in the field in Section 2. Second, Section 3 describes the original mRMR method, and Sections 4 and 5 present our mRMR adaptation to the Arabic summarization task. Section 6 details our experiments in both single- and multi-document summarizations. Finally, we present our paper's conclusions and several interesting perspectives in Section 7.

2. Previous work

Identifying relevant sentences is the key element for extractive summarization systems. Statistical techniques assign a score to each sentence. Computing this score varies from the use of positional- and frequency-based information to the use of topic signatures, abstractive terms and sentence recommendation.

Compared to English document summarization, very few works have been performed for Arabic document summarization. To the best of our knowledge, Douzidia and Lapalme (2004) was the first work in the Arabic summarization field. It uses classical sentence scoring features: sentence position, terms frequency, title words (Luhn, 1958) and cue words (Edmundson, 1969): for example, "تجدر الإشارة إلى" or "we underline" and "وبناء على ما سبق" or "in conclusion" are used to capture sentences in which the author has emphasized. Douzidia and Lapalme (2004) used a weighted linear combination of these features (which is often the case) to score sentences (1). The system uses character level normalization, a light lemmatization (simple prefix and suffix removal) and a rule-based sentence compression component to reduce several indirect discourse parts, such as name substitution.

$$Sc = \alpha_1 Sc_{lead} + \alpha_2 Sc_{title} + \alpha_3 Sc_{cue} + \alpha_4 Sc_{tf.idf} \quad (1)$$

Sobh et al. (2006) used additional features: the sentence and paragraph length, the sum of sentence cosine similarity values with the rest of sentences and some POS-based features: number of infinitives, verbs, *Marfo'at*, and *identified* words and whether the sentence includes a digit or not. Next, the authors apply three classifiers: two basic classifiers (Bayesian, genetic programing) and a combination of these two as a dual one. Among this work's interesting conclusions is the fact that on the basis of an evaluation on 213 articles from "Al Ahram" newspaper, features were classified into three categories: strong, weak and intermediate. Strong features were: the sentence's term weight, length and similarity sum.

Schlesinger et al. (2008) used a rule-based sentence splitter and six-gram tokenization to process Arabic texts. The authors outlined the lack of resources to accomplish these two tasks. Motivated by Douzidia and Lapalme's (2004) good evaluation results, authors use original Arabic texts rather than English machine translated texts, a unigram language model and signature terms to score sentences. Once top ranked sentences are extracted, the system replaced Arabic sentences with the corresponding machine translated (MT) sentences.

El-Haj et al. (2011a) proposed two Arabic text summarization systems: AQBTSS, a query-based Arabic single-document summarizer, and ACBTSS, a concept-based summarizer. The first, AQBTSS, attempts to fit the generated summary to a specific Arabic user's query while the second, ACBTSS, attempts to fit it against a bag-of-words representation of a certain concept. The two systems use a vector space model to score sentences. Interestingly, for the concept-based summarizer, the author dressed a list of 10 concepts with the corresponding most frequent terms. This was defined on the basis of a 10,250 Arabic newspaper articles corpus with approximately 850 documents per concept. On the basis of this work, the KALIMAT 1.0[3] corpus was recently released; it is a free and publically available dataset of 20,291 newspaper articles which

[3] http://sourceforge.net/projects/kalimat/.

fall into six categories: culture, economy, local news, international news, religion, and sports. It contains the corresponding single- and multi-document extractive summaries. A full NER, POS and morphological analysis is also available for all of the dataset articles.

Other works try to *understand* the source text and rely on a deep analysis of the source text. Indeed, Mathkour et al. (2008) built rhetorical structure trees (RS) (Mann and Thompson, 1988; Marcu, 2000) for Arabic texts. They used a list of 11 Arabic rhetorical discourse relations (Condition or "الشرط", Justification or "التعليل", etc.) with their corresponding cue phrases ("إن، إذا، لو", "لأن، بسبب", etc.) to build a binary tree of textual span pairs (nucleus, relation type, and satellite). The nucleus span is therefore more important to the reader than the satellite span, and the summary is generated using the first levels of the RS tree. Obviously, there is more than one tree per text because of the ambiguity of relations and text span attachments. Consequently, various summaries could be generated for the same text. In Al-Sanie et al. (2005), the authors discussed determining more suitable options for the summarization task among possible trees.

Similarly, Mâaloul et al. (2010) identified 19 possible rhetorical relations through a corpus-based analytical study, and some of them have commonalities with Mathkour et al. (2008). Second, they limited the initial relations to a list of the nine most useful relations for the summarization task. Azmi and Al-Thanyyan (2012) proposed a hybrid two-pass summarization, where the first pass uses the RS tree's first levels (Mathkour et al., 2008) to generate a primary summary, while the second pass uses the primary summary to produce a shorter version. The second pass scores sentences within the primary summary using a Douzidia and Lapalme (2004)-like scoring formula (1). The authors affirm that the two-pass summarizer improves the basic RST summarizer.

RST-based summarization techniques rely on a limited number of human-made rhetorical relation patterns and their corresponding cue phrase lexicons. Identifying these relations with an acceptable accuracy is a hard task, even for a native Arabic speaker because of the high semantic ambiguity of the Arabic language. In addition, computational time is not a negligible factor. Indeed, we hardly expect to automatically construct accurately several possible RS trees for a 200-page text in a reasonable time. In fact, computational time is not specific to the Arabic language; it is a drawback that researchers want to overcome in every heavy NLP-based system. Furthermore, it is well known that sentence *informativeness* and *centrality* is not sufficient and that information *novelty* is also an important feature. Indeed, considering information novelty maximizes the summary's *coverage* and avoids *redundancy*.

Recently, a new task has been introduced to the summarization field: multilingual summarization. Here, the goal is to design systems applicable to various languages. Therefore, the system should use very little language-dependent knowledge. This system is different from the early summarization systems Barzilay and McKeown (2005), Marcu, 2000; Radev and McKeown (1998) and was motivated by better results and robustness of statistical techniques. The TAC MultiLing 2011 workshop (Giannakopoulos et al., 2011) strengthened this new trend and made a multilingual dataset of seven languages available, including Arabic. The workshop allowed systems (which were not originally developed for Arabic except for El-Haj et al. (2011b)) to produce multi-document news summaries for seven different languages, including Arabic. A hierarchical latent Dirichlet allocation-based system (Liu et al., 2011) ranked first and the El-Haj et al. (2011b) system ranked third. These efforts continued during the ACL multilingual 2013 workshop (Giannakopoulos, 2013). Alguliev et al. (2011) modeled the sentence relevance and redundancy dilemma as an optimization problem. The maximum coverage and minimum redundant (MCMR) text summarization system computes sentence relevance as its similarity to the document set vector. Sentence redundancy is its similarity to the remaining sentences. MCMR uses two integer linear programing (ILP) algorithms: branch-and-bound and binary particle swarm to select the most relevant and dissimilar sentences with respect to the summaries' size. Later, they use a genetic algorithm and differential evolution (DE) algorithm to solve the optimization problem (Alguliev et al., 2013).

The success of these *foreign* systems on the Arabic language and the lack of robust Arabic NLP tools encouraged us to develop a statistical summarization system for Arabic texts. It uses basic Arabic NLP resources (sentence segmentation, tokenization and root extraction) and attempts to address information diversity without omitting sentence relevance along the sentence scoring and extraction process. In fact, it first uses a discriminant analysis method, mRMR (Peng et al., 2005), to score terms. Second, it uses the resulting mRMR scores to rank sentences according to the discriminant power of their terms. We experiment with our method on single- and multi-document Arabic datasets: EASC and TAC 2011 multilingual datasets under different configurations.

3. Minimum redundancy maximum relevance (mRMR)

Minimum redundancy maximum relevance (mRMR) (Peng et al., 2005) is a discriminant analysis method whose goal is to select a subset of features which best represents the whole space of features. It is based on mutual information (2) between pairs of features, which reflects the level of similarity between them.

$$I(X;Y) = \sum_{y\in Y}\sum_{x\in X} p(x,y)\log\frac{p(x,y)}{p(x)p(y)} \tag{2}$$

In fact, mutual information aims to measure the information quantity that two features share. Therefore, if two features have a high mutual information quantity, we say that they are highly correlated, and consequently, one can replace the other with minimum information loss.

Peng et al. (2005) used mutual information to measure redundancy and relevance at the same time. Indeed, to minimize redundancy, we are interested in finding dissimilar features (with minimum mutual information score), which represent the whole features (with maximum relevance).

Maximizing relevance requires selecting features, which represent at best the entire dataset. The authors accomplished this in two steps:

1. Applying a classification method to find different classes of observations.
2. Then, they opted, again, for a mutual information score between each feature and the classification variable resulted from one.

To use mRMR, we need as an input, a matrix, where each column represents a feature and each row represents the corresponding observations. Every observation belongs to a specific class; thus, each row of the input matrix is attached to a class number c (the value of the classification variable).

$$\begin{bmatrix} c & f_1 & f_2 & & f_n \\ 1 & 1 & -1 & & -1 \\ 1 & -1 & -1 & \cdots\cdots\cdots & 1 \\ 2 & -1 & 1 & & -1 \\ 2 & 1 & 1 & & 1 \\ & \vdots & & \ddots & \vdots \\ & & & \cdots\cdots\cdots & \end{bmatrix}$$

In this approach, we try to select the top m and $m < n$ features with a maximum relevance (mutual information {feature f_i, classification variable c}, and at the same time, with a minimum redundancy (mutual information between two features f_i, f_j).

Example:

If we have two classes, two features and five observations:

$$\begin{vmatrix} c & f_1 & f_2 \\ 1 & -1 & -1 \\ 2 & +1 & +1 \\ 1 & -1 & +1 \\ 2 & +1 & -1 \\ 1 & -1 & -1 \end{vmatrix}$$

At first, f_1 appears to be more relevant than f_2. It is absent (value = −1) in the first class 1 and present (value = 1) in the second class 2 for all five observations, unlike f_2. In fact, mRMR scores confirm this observation: 0.971, 0.00 for f_1, f_2, respectively.

To combine these two values, the authors proposed two methods: maximizing the difference between relevance and redundancy or maximizing the quotient between them. mRMR was applied in the field of bioinformatics (Ding and Peng, 2003), where the idea was to select the genes responsible for a cancer type among others. Classes represented cancer types, and features represented examined genes within a class. According to the authors, the results were very satisfying, sometimes reaching a 97% accuracy threshold.

In the context of text summarization, this method will be especially useful. mRMR is a pure statistical and robust method and will help us to assess diversity contribution compared to redundancy in various contexts: short and long documents, same or different domains, single- or multi-document summaries, and summaries' length. The following section describes our adaptation of the mRMR method to the task of Arabic text summarization.

4. mRMR for Arabic single-document summarization

Terms' frequency is a relevance indicator in the source text and, at the same time, a drawback that every summarization system wants to avoid in the result summaries. To investigate this issue, we propose to use the minimum redundancy and maximum relevance method described above. Here, our goal is to perform extractive summarization of Arabic texts, i.e., sort sentences within a document and keep those that maximize relevance and, at the same time, cover up, at best, information contained in the source document.

Measuring the relevance of a specific sentence is the main novelty in our proposition. Indeed, it depends on the relevance of terms within this sentence. To achieve this, we first use a clustering algorithm to group similar sentences into clusters. The choice of a good clustering method is decisive for the success or failure of mRMR. Accurate clusters will, obviously, lead to a better estimation of term relevance within the source document.

To score terms using mRMR, we first perform a pretreatment on the source text and apply a clustering method to group similar sentences into clusters. Fig. 1 shows different steps that our system performs.

4.1. Text preprocessing

In our system, the sentence is the extraction unit and the term is the scoring unit, and it becomes necessary to perform preprocessing of the source document. Here, we need to run the following classic steps:

A. Sentence splitting;
B. Tokenization;
C. Stop words removal; and
D. Root extraction.

For Arabic texts, these basic NLP tasks are problematic. Indeed, sentence segmentation is a difficult task because of non-capitalization and minimal punctuation (Farghaly and Shaalan, 2009).

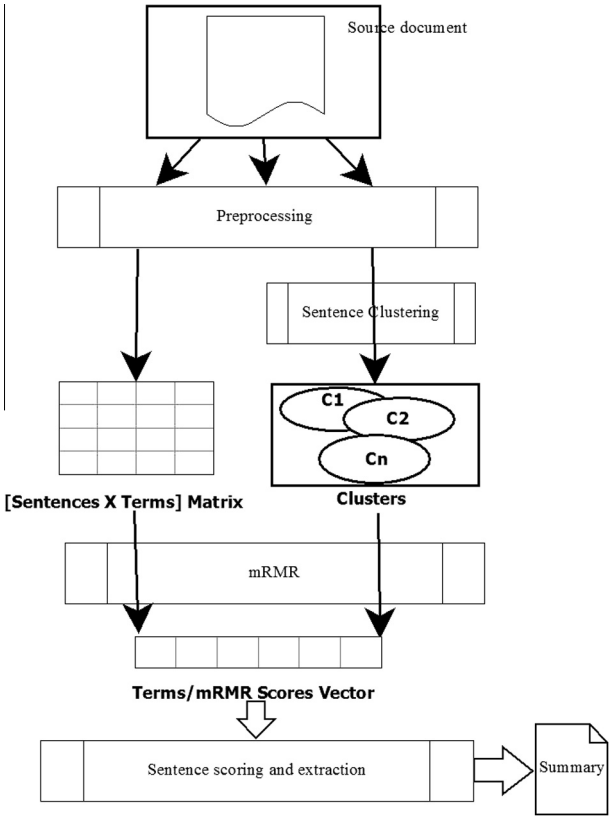

Figure 1 System architecture.

Indeed, it is common to find an entire paragraph without punctuation in Arabic texts. We usually use Arabic coordinators such as "و" *wa* and "ف" *fa* instead of the ".". Unfortunately, as far as we know, there is no Arabic sentence splitter that considers all of these particularities. We rely on a naive "." -based splitter. This choice was forced but appropriate because in the two datasets we used, sentences were separated by ".". We look forward to use a *real* Arabic sentence splitter once it is available.

Farghaly and Shaalan (2009) defined an Arabic word as a string of characters delimited by spaces. With the agglutinative propriety of the Arabic language, it is possible to formulate a complete sentence with as many as four different parts of speech with one Arabic word, "فكتبته" or "So I wrote it" is an example. This makes tokenization and root extraction tasks especially challenging. For tokenization, we use a Stanford Arabic word segmenter (Monroe et al., 2014), which achieves very good precision scores and is particularly fast compared to the well-known MADA system (Habash and Rambow, 2005). Then, we use a list of 168 words defined by Khoja (1999) to remove stop words. For root extraction, we use an updated version of the root-based approach developed by Khoja (1999)[4].

Finally, we project the sentence vector (resulting from the first step: sentence splitting) onto the term vector (resulting from the last step: root extraction) to obtain, at last, the [Sentences × Terms] matrix in which each entry $[i, j]$ corresponds to Term_j frequency in the sentence i.

Sentence	Term_1	Term_2
$S1$	0	0
$S2$	1	1
$S3$	0	2
$S4$	1	0
$S5$	0	0

4.2. Sentence clustering

Once the [Sentences × Terms] matrix is computed and before applying mRMR, we need to perform sentence clustering. Given the [Sentences × Terms] matrix, we compute the similarity matrix [Sentences × Terms] where each value $[i, j]$ corresponds, for example, to the widely used method: cosine similarity between couples of sentences X and Y (3):

$$\cos(X, Y) = \frac{\sum_i x_i y_i}{\sqrt{\sum_i (x_i)^2} \cdot \sqrt{\sum_i (y_i)^2}} \tag{3}$$

The similarity or the distance matrix (Distance $[i, j] = 1 -$ Similarity $[i, j]$, $i > j$) is the input to the clustering algorithm. Here, we use the hierarchical clustering HCLUST. Of course, any good clustering algorithm could be used instead. We will discuss, further, the clustering algorithm and the number cluster impact on the resulting summaries (§ Section 6).

Once clustering is performed, the [Sentences × Terms] matrix will be augmented by a new column: the class number of each sentence. The mRMR scoring algorithm uses this final matrix as an input.

[4] http://sourceforge.net/projects/arabicstemmer/.

Sentence	Class	Term_1	Term_2
$S1$	1	0	0
$S2$	2	1	1
$S3$	2	0	2
$S4$	1	1	0
$S5$	1	0	0

4.3. Scoring terms using mRMR

mRMR scores feature on the basis of how much discriminant information they hold. In summarization, we are interested in highly discriminant terms, which allow us to select a specific sentence and not the other sentences. Finding this set of terms is what our adaptation of mRMR accomplishes.

In Peng et al. (2005), the authors proposed to compute the relevance as how much the feature's variation follows the classes' variation. For a term's relevance, it is the discriminating power of a specific term within classes, i.e., the more a term's frequency varies significantly through classes, the more it is discriminant. It is actually better to think about the opposite case; if a term has the same frequency mean (or is close to the term's frequency mean in all classes), it is not quite as interesting, and consequently, will receive a low relevance score.

Formally, we use the F-test to compute the relevance score:

$$F(t, h) = \frac{\left[\sum_k n_k (\bar{t}_k - \bar{t})/(K-1)\right]}{\sigma^2} \tag{4}$$

where n_k is the number of sentences in class k, $\bar{t}_k$ is the mean value of t within class k, $\bar{t}$ is the mean within all of the K classes and σ is the pooled variance.

If we take, as an example, a matrix of two terms with the following mean frequencies:

Example:

Class	Term_1	Term_2
1	2	2
2	0	2
3	5	0

First, Term_1 seems to be more discriminant among different classes than Term_2. We believe that highly discriminant terms will highlight discriminant sentences, which mark idea changes.

The question that arises here is: Is relevance as the discriminating power of terms sufficient for the summarization task? In fact, if two terms share very close relevance scores, we should be able to sort them on the basis of their redundancy scores.

Term's t_i redundancy is the mean of its mutual information with all other terms; if a term has a high redundancy score, i.e., shares an important amount of information with the rest of the terms.

Thus, we compute a term's mutual information as follows:

$$\text{Redundancy}(T_i) = \frac{1}{|S|} \sum_{j \in S} I(T_i; T_j) \tag{5}$$

Let us take a closer look on the meaning of low/high mutual information between couples of terms; if two terms share high mutual information, this shows how much one term

attracts the other, i.e., they tend to appear at the same time within sentences. We use the Pearson correlation (6) to compute this value.

$$C(T_i, T_j) = \frac{\sum_k (T_{i,k} - \bar{T}_i).(T_{j,k} - \bar{T}_J)}{\sqrt{\sum_k (T_{i,k} - \bar{T}_i)^2}.\sqrt{\sum_k (T_{j,k} - \bar{T}_J)^2}} \quad (6)$$

Therefore, if a term has a high redundancy score, which means that it attracts many terms, this will reduce its discriminating power. In the original mRMR method, the others proposed to combine these two values in two ways and perform an incremental search for top n features:

FCD for *F*-test Correlation Difference $\max_{i \in \Omega_s}$

$$\times \left[F(i,h) - \frac{1}{|S|} \sum_{j \in S} |c(i,j)| \right] \quad (7)$$

FCQ for *F*-test Correlation Quotient $max_{i \in \Omega_s}$

$$\times \left[F(i,h) / \frac{1}{|S|} \sum_{j \in S} |c(i,j)| \right] \quad (8)$$

Finally, we succeed in finding a set of terms, which describe the best sentences' clusters and, at the same time, does not attract many terms. This set of terms will guide us to find the most relevant sentences.

The main contribution of our new term scoring method is that the term is the central element of our summarization method instead of the sentence. Once we have the set of terms, we can use them to score sentences. The result of this step is a documents' term vector with the corresponding adapted mRMR scores:

$$V_{\text{mRMR}} = (w_{t1}, w_{t2}, \ldots\ldots\ldots\ldots, w_{tn}) \quad (9)$$

The next paragraph details our sentence scoring method and extraction algorithm based on the new mRMR weights.

4.4. Scoring sentences using adapted mRMR scores

Based on the new mRMR scores, we can use several sentence scoring strategies:

(a) The mRMR weights sum

$$\text{Score}(S) = \sum_{i=1,n_s} w_{t_i}, t_i S \quad (10)$$

(b) The Sentence/V_{mRMR} Cosine similarity

$$\text{Score}(S) = |\text{Cosine\_Similarity}(V_s, V_{mRMR})|$$

(c) Select m mRMR terms, $m < n$ and score S using one of the two strategies above.

The first strategy tends to favor longer sentences; the second strategy includes a normalization factor so that longer sentences do not obtain more weight. The third strategy would be interesting if we want to focus on a specific term subset or the 10 most discriminant terms, for example.

4.5. Sentence extraction algorithm

The last step that our summarization system performs is to extract the top n sentences depending on the required summary size (compression ratio, number of sentences, and number of characters). This extraction could be a simple reverse sorting or a recursive process.

We propose a novel extraction algorithm. It takes into account terms within already selected sentences to compute the score of the next sentence to be included in the summary. Let V_{mRMR} be the mRMR term vector resulting from step 3 and $V_s = \{(s_i, w_i), i \in S\}$ be the vector of all sentences associated with their initial scores resulting from step 4. The main idea is to decrease the discriminant term weight along selecting sentences in which they appear. We propose two decreasing speeds: rapid and slow. For the first speed, rapid decrease, the already included term weights are set to zero. This appears to be suitable for very short summaries and allows us to select the maximum information quickly. The second speed, slow decrease, decreases the already included mRMR terms weights progressively, depending on the just selected sentence similarity s_j to V_{mRMR}: sim_j.

Input: $V_{mRMR} = (w_{t1}, w_{t2}, \ldots\ldots\ldots\ldots, w_{tn})$
$S = \{(s_i, sim_i), i \in S\}$
$Size_R$

1. $R = \Phi$
2. Select s_j, $sim_j = Max(sim_i, i \in S)$
   a. $R = R \bigcup \{s_j\}$
3. Update V_{AmRMR}
   a. $T = V_{mRMR} \cap T_{s_j}$
   b. For each $t_k \in T$, update V_{mRMR} weights
   – If selection method is slow decrease
   $w'_k = w_k - sim_j{}^* w_k$
   If selection method is rapid decrease
   $w'_k = 0$
4. Update S
   a. $S = S - s_j$
   b. For each $s_k \in S$, update sentences scores
   Score $(s_k)' = |Sim(V_{s_k}, NewV_{mRMR})|$
5. Go to 2 while $(Size(R) < Size_R$ and $\exists\ w_t > 0)$

Output: Summary R

5. mRMR for multi-document summarization

In the multi-document summarization task, sentence scoring is even more difficult because of the high probability of the cross-sentence informational subsumption CIDR, if the documents discuss the same topic. CIDR reflects the fact that certain sentences repeat some of the information present in other sentences. Radev et al. (2004) proposed to compute the CIDR as the number of terms two sentences have in common, normalized by the length of each one.

Because we use terms as the central scoring elements, our adapted mRMR could be easily used to produce multi-document summaries. Sentence position is not incorporated in the scoring formula, and we can thus use one of the clustering methods:

(a) Each document corresponds to a cluster;
(b) Consider the bag-of-sentences model and apply a clustering algorithm (Naive Bayes, SVM, and HCLUST) to produce clusters and define the classification variable.

The following steps, terms and sentences scoring, remain unchanged and could be performed similar to the single-document summarization task.

6. Experiments

6.1. Datasets

Because we evaluate our system for both Arabic single- and multi-document summarizations, we used two datasets: Essex Arabic summaries corpus (EASC) and MultiLing Pilot 2011 dataset.

➢ Essex Arabic Summaries Corpus (EASC)

For single-document summarization, we use the Essex Arabic summaries corpus (El-Haj et al., 2010). The dataset contains 153 documents from Wikipedia, AlRai and AlWatan newspapers[5]. The dataset contains 10 main topics: art and music, education, environment, finance, health, politics, religion, science and technology, sports and tourism.

For each document, five model extractive summaries are available. These model summaries were generated by native Arabic speakers using Mechanical Turk[6]. Users were asked to select sentences focusing on the main idea of the source document. The model summaries size does not exceed 50% of the source document's size. The dataset is available in two encodings: UTF-8 and ISO-Arabic.

➢ MultiLing Pilot 2011 Dataset

For the multi-document summarization task, we used the Text Analysis Conference (TAC) 2011 MultiLing Pilot dataset (Giannakopoulos et al., 2011). It is a parallel multilingual corpus of seven languages: Arabic, Czech, English, French, Greek, Hebrew and Hindi. The creation of the corpus started by gathering an English corpus, and it contains 10 document sets with 10 documents for each set. The original news articles were extracted from the WikiNews[7] website. Each document set describes one event sequence: the 2005 London bombing or the Indian Ocean Tsunami. Texts in other languages, including Arabic, have been translated by native speakers of each language.

For each document set, three model summaries are provided by fluent speakers of each language (native speakers in most cases). Each model summary size is between 240 and 250 words. The dataset is available in UTF-8 encoding.

6.2. Evaluation metrics

To evaluate our system's performance, we use a well-known automatic evaluation method: recall-oriented understudy for gisting evaluation (ROUGE), which we adapted to the Arabic texts.

➢ ROUGE

The ROUGE method (Lin, 2004) has been used in DUC conferences. ROUGE allows us to make an intrinsic evaluation of text summaries against human-made abstracts. It includes five measures: ROUGE-N, ROUGE-L, ROUGE-W, ROUGE-S and ROUGE-SU.

ROUGE-N compares the N-grams of two summaries and counts the number of matches between these two summaries. It can be computed using the following formula:

$$\mathrm{ROUGE}-(\mathrm{N}) = \frac{\sum_{S \in Summ_{ref}} \sum_{N-gram \in S} Count_{match}(N-gram)}{\sum_{S \in Summ_{ref}} \sum_{N-gram \in S} Count(N-gram)} \tag{11}$$

where N is the length of the $N-gram$ and $Count_{match}(N-gram)$ is the maximum number of N-grams co-occurring in a candidate summary and a set of reference summaries ($Summ_{ref}$). $Count(N-gram)$ is the count of $N-grams$ in the reference summary.

ROUGE-S (skip-bigram co-occurrence) is only an extension of ROUGE-N. It is calculated the same as ROUGE-2 but uses skip-bigrams instead of adjacent bigrams. A skip-bigram, as defined in Lin (2004), is any pair of words in their sentence order, allowing for arbitrary gaps. ROUGE-SU is an extension of ROUGE-S, adding unigram as the counting unit, which is a weighted average between ROUGE-S and ROUGE-1.

In our evaluation, we used two metrics of ROUGE, which were used in DUC 2007: ROUGE-1 and ROUGE-2. Because we compare Arabic texts, we deliberately disable the use of porter stemmer; this will lead to a word-by-word comparison and not a stem-by-stem comparison. Thus, recall will obviously decrease. We also modified the original Perl script at the tokenization level to support Arabic script characters instead of Latin script characters.

ROUGE-1 (ROUGE-2) compares the unigram (bigram) overlap between the system summary and the human abstracts. For each of these metrics, we used the recall (R), precision (P), and F1-score (F1), given in the following formula:

$$R = \frac{|G_{ref} \cap G_{can}|}{|G_{ref}|}, P = \frac{|G_{ref} \cap G_{can}|}{|G_{can}|}, F1 = \frac{2PR}{P+R} \tag{12}$$

where G_{ref} includes the grams of reference summary and G_{can} includes the grams of candidate summary.

6.3. Experiment setup

Our experiments tend to achieve the following purposes:

- Evaluate the use of a statistical method on the Arabic texts summarization;
- Evaluate the discriminant analysis contribution to the summarization task;
- Evaluate the impact of the clustering's precision level on the produced summaries quality;
- Comparing our system's single- and multi-document summaries to baselines and reference summaries;
- Comparing our system to the TAC 2011 MultiLing workshop participating systems.

Note that we used the well-known cosine measure to compute sentence similarity and the hierarchical clustering

[5] http://www.wikipedia.org/, http://www.alrai.com/, http://www.alwatan.com.sa.
[6] http://www.mturk.com/.
[7] http://www.wikinews.org/.

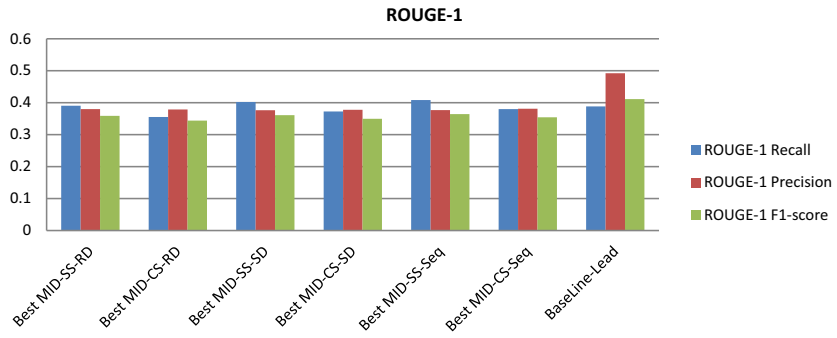

Figure 2 Arabic single-document ROUGE-1 results.

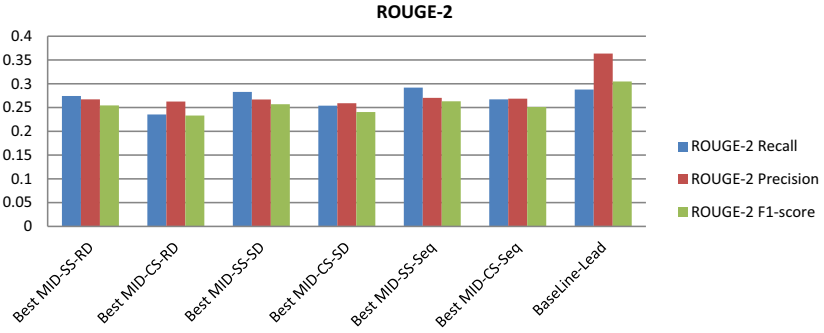

Figure 3 Arabic single-document ROUGE-2 results.

(HCLUST) to perform sentence clustering. Of course, any other similarity/clustering method could be used.

To achieve these purposes, the experiment steps we pursued are as follows:

- Merge texts and produce one text per document set for the multi-document summarization task;
- Splitting text into sentences;
- Generating the [Sentences × Terms] Matrix;
- Computing the similarity/distance matrix [Sentences × Sentences];
- Perform sentence hierarchical clustering by varying the number of classes: $C = 2, 4, 6, 8$, and 10.
- Application of the mRMR method on different configurations;
- Selection of top n sentences and adjusting the size of the summary;
- Computing ROUGE-1 and ROUGE-2 scores of our mRMR summaries and comparing them against reference summaries;
- Comparing ROUGE-1 and ROUGE-2 scores of our mRMR summaries to the baselines and available peer systems' results.

6.4. Results and discussion

We ran our system under three different configurations: RD for rapid decrease, SD for slow decrease and Seq for sequential, i.e., simple top n sentences. For each configuration, we used two sentence scoring methods: Simple Sum -SS- of terms' mRMR scores and Cosine Similarity -CS- between the sentence vector and the mRMR vector. This made a total of six runs. For each run, we took best ROUGE scores corresponding to a certain number of clusters. For example, the SS-SD run for the best rouge score was recorded for 10 clusters. Best scores were generally obtained for 8 to 10 clusters. Note that the -MID- refers to the use of the mutual information difference for mRMR scoring.

➢ Arabic single-document summaries

Figs. 2 and 3 show ROUGE-1 and ROUGE-2 scores for our system against model summaries. Unfortunately, we did not find any peer summary results. Thus, we compare our summaries to the baseline lead sentences.

The baseline lead sentences extract the first n sentences of each document; n is equal to the model summary's size. For the EASC dataset, we have five model summaries per document. Consequently, for each document, we produce five baseline summaries; then, we compute the average of their ROUGE-1 and ROUGE-2 scores. Because of the nature of dataset documents (news and Wikipedia articles), it turns out to be a strong baseline to beat. Indeed, news articles follow the inverted pyramid style [8]and address the most important information first. Wikipedia articles usually begin with a lead section[9], which serves as an introduction to the article and a summary of its most important aspects.

[8] http://en.wikipedia.org/wiki/Inverted_pyramid.
[9] http://en.wikipedia.org/wiki/Wikipedia:Lead_section.

The results show that compared to the baseline, the MID-SS-Seq configuration (mutual information difference, simple sum and top n sentences) outperforms the baseline in terms of recall. Precision scores are lower but are still comparable the baseline scores. Note that our system does not use any positional feature, which is the case for the baseline.

The following sample displays our system summaries (peers) compared to the model and baseline summaries. We can fairly observe that the baseline and peer summaries are more relevant than the model summaries.

Model Summary D0101.M.250.A.1.A

[1] له الفضل الاعظم في تطوير الموسيقى الكلاسيكية.
[2] بدا بيتهوفن يفقد سمعه في الثلاثينيات من عمره الا ان ذلك لم يؤثر على انتاجه الذي ازداد في تلك الفترة وتميز بالابداع.
[3] اتسعت شهرته كعازف بيانو في سن مبكرة، ثم زاد انتاجه وذاع صيته كمؤلف موسيقى.
[4] في 1789 م تحقق حلمه اخيراً، فقد ارسله حاكم بون الى فيينا، وهناك تتلمذ على يد هايدن.
[5] فجاءت رسالته الى العالم كل البشر سيصبحون اخوة.

Model Summary D0101.M.250.A.1.B

[1] يعتبر من ابرز عباقرة الموسيقى في جميع العصور، وابدع اعمالاً موسيقية خالدة.
[2] تشمل مؤلفاته للاوركسترا تسعة سيمفونيات وخمس مقطوعات موسيقية على البيانو ومقطوعة على الكمان.
[3] بدا بيتهوفن يفقد سمعه في الثلاثينيات من عمره الا ان ذلك لم يؤثر على انتاجه الذي ازداد في تلك الفترة وتميز بالابداع.
[4] من اجمل اعماله السمفونية الخامسة والسادسة والتاسعة.
[5] عانى بيتهوفن كثيراً في حياته، عائلياً وصحياً، فبالرغم من ان اباه هو معلمه الاول الذي وجه اهتمامه للموسيقى ولقنه العزف على البيانو والكمان، الا انه لم يكن الاب المثالي، فقد كان مدمناً للكحول، كما ان والدته توفيت وهو في السابعة عشر من عمره بعد صراع طويل مع.
[6] فهل كان التاليف الموسيقي هو نوع من انواع العلاج والتغلب على المشاكل بالنسبة لبيتهوفن.
[7] ولكن بيتهوفن، صاحب الالحان واجه بعض الخلافات مع معلمه، وعندما سافر هايدن الى لندن، تحول بيتهوفن الى معلمين اخرين مثل ساليري وشينك والبريشتبيرجر.
[8] كما انه احدث الكثير من التغييرات في الموسيقى، وادخل الغناء والكلمات في سيمفونيته التاسعة.

BaseLine – Lead Sentences

[1]لودفيج فان بيتهوفن مؤلف موسيقي ألماني ولد عام 1770 م في مدينة بون
[2] يعتبر من ابرز عباقرة الموسيقى في جميع العصور، وأبدع أعمالاً موسيقية خالدة
[3] له الفضل الأعظم في تطوير الموسيقى الكلاسيكية
[4] قدم أول عمل موسيقي وعمره 8 سنوات
[5] تشمل مؤلفاته للأوركسترا تسعة سيمفونيات وخمس مقطوعات موسيقية على البيانو ومقطوعة على الكمان

Peer MID-SS-Seq –A

[1] لودفيج فان بيتهوفن مؤلف موسيقي ألماني ولد عام 1770 م في مدينة بون
[2] ظهر تميزه الموسيقي منذ صغره، فنشرت أولى أعماله وهو في الثانية عشرمن عمره عام 1783 م
[3] تشمل مؤلفاته للأوركسترا تسعة سيمفونيات وخمس مقطوعات موسيقية على البيانو ومقطوعة على الكمان
[4] كما ألف العديد من المقطوعات الموسيقية كمقدمات للأوبرا
[5] فهل كان التأليف الموسيقي هو نوع من أنواع العلاج والتغلب على المشاكل بالنسبة لبيتهوفن

Peer MID-SS-Seq –B

[1] لودفيج فان بيتهوفن مؤلف موسيقي ألماني ولد عام 1770 م في مدينة بون
[2] ظهر تميزه الموسيقي منذ صغره، فنشرت أولى أعماله وهو في الثانية عشرمن عمره عام 1783 م
[3] شهدت مدينة بون الألمانية ميلاد الفنان العبقري لودفج فان بيتهوفن في 16 ديسمبر عام 1770، وتم تعميده في 17 ديسمبر 1770
[4] قدم أول عمل موسيقي وعمره 8 سنوات
[5] تشمل مؤلفاته للأوركسترا تسعة سيمفونيات وخمس مقطوعات موسيقية على البيانو ومقطوعة على الكمان
[6] كما ألف العديد من المقطوعات الموسيقية كمقدمات للأوبرا
[7] يعتبر من أبرز عباقرة الموسيقى في جميع العصور، وأبدع أعمالاً موسيقية خالدة
[8]فهل كان التأليف الموسيقي هو نوع من أنواع العلاج والتغلب على المشاكل بالنسبة لبيتهوفن

From these observations, multiple questions arise: are model summaries "ideal" enough to be considered as reference summaries? Does automatic evaluation reflect the real quality of peer summaries? We believe that a combined manual/automatic evaluation will lead to better evaluation.

➢ Arabic multi-document summaries

The following figures present the ROUGE scores obtained by our system against the baseline, topline and other TAC 2011 multilingual workshop participating systems scores. The BaseLine system -ID9- uses as a summary the text most similar to the documents' set centroid (Radev et al., 2004). Topline -ID10- uses the first model summaries as a *given* to generate a graph-based representation of the best possible summary. Next, it uses a genetic algorithm to select from random summaries of the documents' set, the one that mostly matches the model summaries graph using the MeMoG score (Giannakopoulos and Karkaletsis, 2010).

For ROUGE-1 recall scores (number of words the candidate and model summaries have in common), our system RD, SD and Seq configurations have average scores compared to the Baseline, Topline and seven other participating systems. Our system's F-scores are among the best peer systems' (ID3, ID4 and ID8) scores. Note that our system's recall scores need improvement (See Fig. 4).

Figs. 5 displays the ROUGE-2 results; the best results were reported by the Topline system. Our system's recall is lower than the peer systems' recall. Precision scores were better graded and led to better F-scores.

➢ Experimenting with Arabic-English cross-lingual summarization

Cross-lingual summarization is the task of producing summaries in different languages from the source document summaries: producing English summaries for Arabic documents and vice versa is an example. This transition from one language to another needs, obviously, the use of machine translation (MT).

MT can be incorporated into a summarization system in different ways: Translate source documents then summarize (Evans et al., 2005) or predict the translation quality of each sentence instead of translating it and then select sentences according to their informativeness and translation quality prediction (Boudin et al., 2011; Wan et al., 2010). Consequently, a highly informative but difficult-to-translate sentence may not appear in the summary and vice versa. MT quality prediction aims to reduce the negative impact of MT on the produced summary coherence and reading fluency.

Here, we take advantage of the TAC 2011 multilingual parallel corpora to assess the impact of translation to the English-Arabic cross-lingual summarization. Indeed, we first translate our system's Arabic summaries to English using the Google Translation service[10]. Second, we compare translated summaries against model English summaries using the ROUGE

[10] https://translate.google.com/.

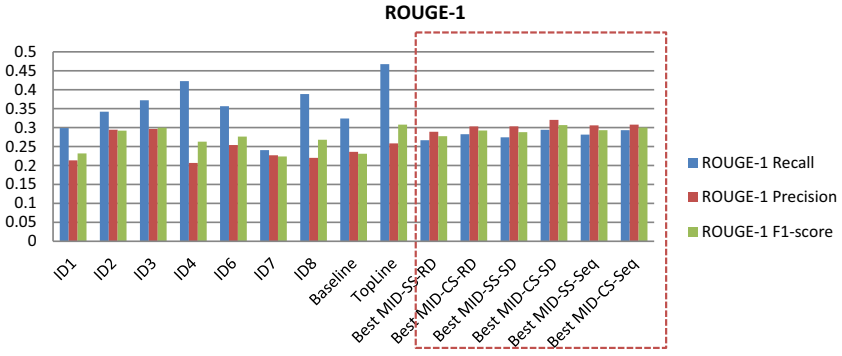

Figure 4 Arabic multi-document summarization ROUGE-1 results.

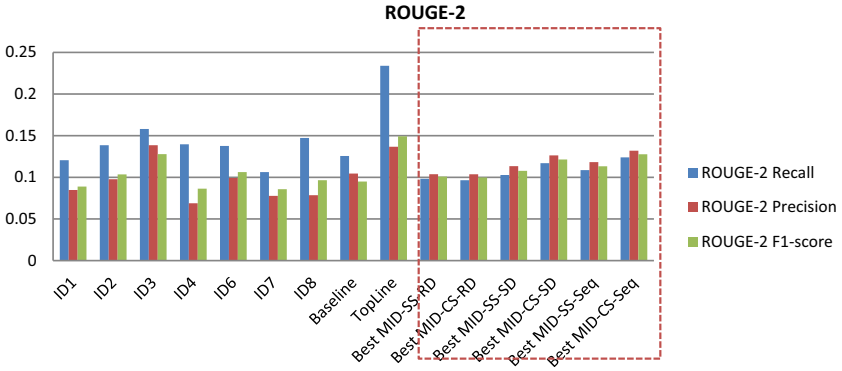

Figure 5 Arabic multi-document summarization ROUGE-2 results.

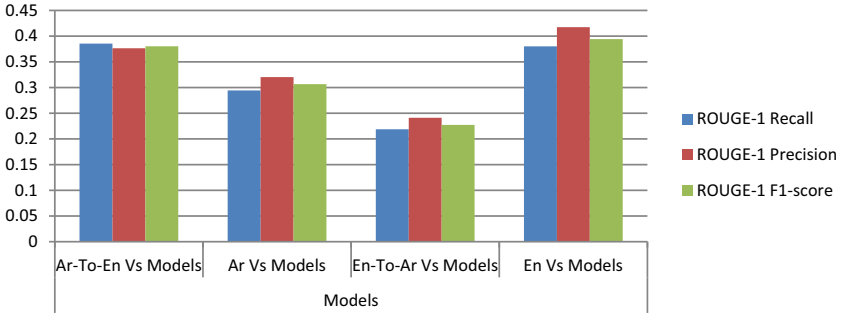

Figure 6 ROUGE-1 results for Arabic-English cross-lingual multi-document summarization.

metric. We have also performed the opposite: summarizing English documents using our system then translating them to Arabic and comparing them against Arabic model summaries. Fig. 6 shows the ROUGE-1 results.

We observe that when Arabic is the target language, ROUGE-1 scores decrease considerably (by more than one-half), which is not the case when English is the target language. In Fig. 7, the ROUGE-2 results confirm this observation.

Hence, for the same translator (Google Translate, which is among the best translators in the market), translating from Arabic to English is considerably better than the opposite. In fact, this observation emphasizes how much the MT quality could affect the summary quality, at least in term of informativeness. In terms of reading fluency and coherence, we believe that they are even more affected. Improvements are needed for the to-Arabic translation field.

7. Conclusions and future work

In this paper, we introduced a novel Arabic single- and multi-document summarization method based on automatic sentence clustering and an adapted discriminant analysis method: mRMR. Our adapted version takes advantage of terms' discriminant power to score sentences and put forward the most salient sentences. We have proposed different configurations on how to use our scoring method, depending on the requested

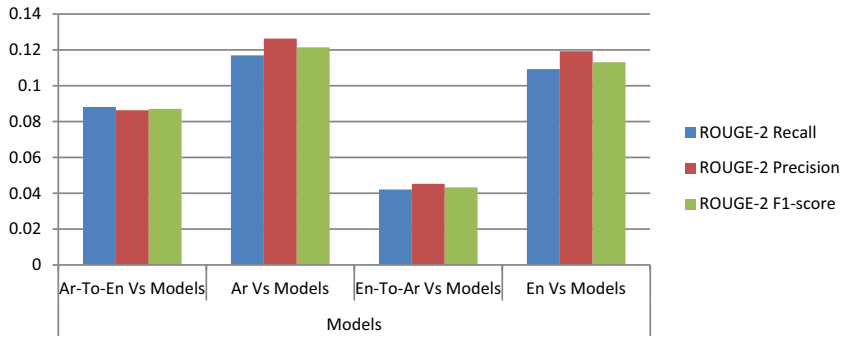

Figure 7 ROUGE-2 results for Arabic-English cross-lingual multi-document summarization.

summary size (Very Short: speed decrease, Short: slow decrease). We have also noted the sentence clustering benefit with our adapted mRMR method. Our method uses minimum language-dependent processing: Only at the root extraction level and does not use any structural or domain-dependent features and was, therefore, successfully used to summarize Arabic texts.

The evaluation results were promising in terms of overall score. We have evaluated our system's performance in both single- and multi-document summarization tasks. For single-document summarization tasks, our method outperforms the strong baseline: lead sentences in terms of recall. For multi-document summarization tasks, we compared our system to other systems: the TAC 2011 multilingual workshop participating systems, and the results were acceptable. Note that we followed an automatic evaluation procedure and compared our summaries with human-made summaries using the ROUGE method.

However, *serious* questions arose and outlined some of the automatic evaluation's shortcomings. In fact, we believe that some model summaries were not sufficiently "ideal" to be considered as reference summaries and even if it was the case, the ROUGE method is not sufficient and should be balanced with a manual evaluation. We look forward to performing a manual evaluation to assess the significance differences between systems and come closer to a *fair* enough evaluation.

The lack of basic Arabic NLP tools was also problematic. We used naïve solutions for sentence splitting and tokenization tasks. The root extraction algorithm generated an important silence and biased, in some cases, the sentence similarity measure. Next, we wanted to investigate the integration of some interesting state of the art tokenization tools, such as Attia (2007) and Habash and Rambow (2005). We also wanted to investigate the use of a light stemmer instead of a root-based stemmer to address the sentence similarity computation.

At this stage, the main goal of our summarization method is the identification and extraction of relevant sentences from a set of Arabic documents. This type of summarization has the advantage of extracting complete grammatically correct sentences. However, this automatically leads to a coherence problem in the produced summaries. Improvements should be performed, and user studies could be performed to evaluate coherence and reading fluency.

For multi-document summarization tasks, sentence reordering is also an open problem and was not considered in the present paper. In the future, we intend to conduct experiments on other domains (literature, etc.) and investigate the cross-lingual task: summarizing Arabic documents in other languages: English and French and vice versa.

References

Al-Sanie, W., Touir, A., Mathkour, H. (2005). Towards a suitable rhetorical representation for Arabic text summarization. In Kotsis, G., Taniar, D., Bressan, S., Ibrahim, I.K., Mokhtar, S. (Eds.), iiWAS, vol. 196, Austrian Computer Society, pp. 535–542.

Alguliev, R.M., Aliguliyev, R.M., Hajirahimova, M.S., Mehdiyev, C. A., 2011. MCMR: maximum coverage and minimum redundant text summarization model. Expert Syst. Appl. 38 (12), 14514–14522.

Alguliev, R.M., Aliguliyev, R.M., Isazade, N.R., 2013. MR&MR-sum: maximum relevance and minimum redundancy document summarization model. Int J Inf Technol Decis Making 12 (3), 361–393.

Attia, M.A., 2007. Arabic tokenization system. In: Proceedings of the 2007 Workshop on Computational Approaches to Semitic Languages: Common Issues and Resources. Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 65–72.

Azmi, A.M., Al-Thanyyan, S., 2012. A text summarizer for Arabic. Comput. Speech Lang. 26 (4), 260–273.

Barzilay, R., McKeown, K.R., 2005a. Sentence fusion for multidocument news summarization. Comput. Linguist. 31 (3), 297–328.

Boudin, F., Huet, S., Torres-Moreno, J.-M., 2011. A graph-based approach to cross-language multi-document summarization. Polibits 43, 113–118.

Ding, C., Peng, H. (2003). Minimum redundancy feature selection from microarray gene expression data. Int. J. Bioinform. Comput. Biol., pp. 523–529.

Douzidia, F.S., Lapalme, G., 2004. Lakhas, an Arabic summarization system. Proceedings of the Document Understanding Conference (DUC2004).

Edmundson, H.P., 1969. New methods in automatic extracting. J. ACM 16 (2), 264–285.

El-Haj, M., Kruschwitz, U., Fox, C. (2010). Using mechanical Turk to create a corpus of Arabic summaries. Proceedings of the Seventh Conference on International Language Resources and Evaluation.

El-Haj, M., Kruschwitz, U., Fox, C. (2011a). Experimenting with automatic text summarisation for Arabic. In: Z. Vetulani (Ed.), Human Language Technology. Challenges for Computer Science and Linguistics, Springer, Berlin Heidelberg, pp. 490–499.

El-Haj, M., Kruschwitz, U., Fox, C. (2011b). University of Essex at the TAC 2011 multilingual summarisation pilot. Proceedings of the Text Analysis Conference (TAC2011).

Evans, D.K., Mckeown, K., Klavans, J.L. (2005). Similarity-based Multilingual Multi-Document Summarization. IEEE Transactions on Information Theory, p. 49.

Farghaly, A., & Shaalan, K. (2009). Arabic natural language processing: challenges and solutions. ACM Transactions on Asian Language Information Processing, 8(4), 14:1–14:22.

Giannakopoulos, G., 2013. Multi-Document Multilingual Summarization and Evaluation Tracks in ACL 2013 MultiLing Workshop. Association for Computational Linguistics, Sofia, Bulgaria, pp. 20–28.

Giannakopoulos, G., El-Haj, M., Favre, B., Litvak, M., Steinberger, J., Varma, V. (2011). TAC 2011 MultiLing pilot overview. Proceedings of the Text Analysis Conference (TAC).

Giannakopoulos, G., Karkaletsis, V. (2010). Summarization system evaluation variations based on n-gram graphs. Proceedings of the Text Analysis Conference (TAC2010).

Habash, N., Rambow, O., 2005. Arabic tokenization, part-of-speech tagging and morphological disambiguation in one fell swoop. In: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 573–580.

Khoja, S., 1999. Stemming Arabic text. Computing Department, Lancaster University, Lancaster, UK.

Lin, C. (2004). Rouge: A Package for Automatic Evaluation of Summaries. In Moens, M., Szpakowicz, S. (Eds.), Presented at the Text Summarization Branches Out: Proceedings of the ACL-04 Workshop, Association for Computational Linguistics, pp. 74–81.

Liu, H., Liu, P., Wei, H., Li, L. (2011). The CIST summarization system at TAC 2011. Proceedings of the Text Analysis Conference (TAC2011).

Luhn, H.P., 1958. The automatic creation of literature abstracts. IBM J. Res. Dev. 2 (2), 159–165.

Mâaloul, M. H., Keskes, I., Hadrich Belguith, L., Blache, P. (2010). Automatic summarization of Arabic texts based on RST Technique. In Proceedings of 12th International Conference on Enterprise Information Systems (ICEIS'2010)12th International Conference on Enterprise Information Systems (ICEIS'2010) vol. 2, Portugal. pp. 434–437).

Mann, W.C., Thompson, S.A., 1988. Rhetorical structure theory: toward a functional theory of text organization. Text – Interdiscip. J. Study Discourse 8 (3).

Marcu, D., 2000. The Theory and Practice of Discourse Parsing and Summarization. MIT Press, Cambridge, MA, USA.

Marcu, D.C. (1998). The rhetorical parsing, summarization, and generation of natural language texts. University of Toronto, Toronto, Ontorio, Canada, Canada.

Mathkour, H.I., Touir, A.A., Al-Sanea, W.A., 2008. Parsing Arabic texts using rhetorical structure theory. J. Comput. Sci. 4 (9), 713–720.

Monroe W., Green S., Manning C.D. (2014). Word Segmentation of Informal Arabic with Domain Adaptation. ACL, Short Papers.

Peng, H., Long, F., Ding, C., 2005. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. IEEE Trans. Pattern Anal. Mach. Intell. 27 (8), 1226–1238.

Radev, D.R., Jing, H., Styś, M., Tam, D., 2004. Centroid-based summarization of multiple documents. Inf. Process. Manage. 40 (6), 919–938.

Radev, D.R., McKeown, K.R., 1998. Generating natural language summaries from multiple on-line sources. Comput. Linguist. 24 (3), 470–500.

Schlesinger, J.D., O'Leary, D.P., Conroy, J.M., 2008. Arabic/English multi-document summarization with CLASSY—the past and the future. In: Gelbukh, A. (Ed.), Computational Linguistics and Intelligent Text Processing. Springer, Berlin Heidelberg, pp. 568–581.

Sobh, I., Darwish, N., Fayek, M. (2006). An optimized dual classification system for Arabic extractive generic text summarization. Proceedings of the 7th Conf. on Language English, ESLEC, 149–154.

Wan, X., Li, H., Xiao, J., 2010. Cross-language document summarization based on machine translation quality prediction. In: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 917–926.