SAP: Standard Arabic Profiling Toolset for Textual Analysis

Khalid M. O. Nahar, Ahmed F. Al Eroud, Malek Barahoush, and Abdallah M Al-Akhras

***Abstract*—This paper defines a Standard Arabic Profiling (SAP) toolset that helps researchers for textual analysis and comparing between different Arabic corpora. Since tools for Arabic language are needed, we present the SAP toolset to simplify the textual analysis process. The approach consists of three profilers: The Part of Speech (POS) profiler that gives statistical analysis for a given document, vocabulary profiler which provides user with an indication out the vocabulary used in a document with reference to Open Source Arabic Corpus (OSAC) of two news agencies (CNN and BBC). The process is accomplished by computing similarity between documents and corpus using Log likelihood measure. Lastly the newly added profiler is the Readability profiler which is used to 1) assess the readability level for a document according to Flesch Reading Ease Readability Formula, and 2) measure the simplicity and ambiguity levels of the document. We described the current part-of-speech for this toolset and how we can extend its functionality to embrace vocabulary and readability profiling.**

***Index Terms*—Arabic natural language processing, part-of-speech tagging (POST), text analysis, software.**

I. Introduction

Research in natural language processing (NLP) has witnessed a rapid progress since 1950. Research in the area of NLP focuses on processing the written text, therefore, it addresses practical applications for the written text. This includes, but is not limited to, opinion mining, information extractions and text summarization. This growth is due to the huge web contents being published. It is important to notice that the new trend in NLP is to apply compositional rather than lexical semantics, leading to the so-called next-generation next narrative-based NLP technology [1].

The Arabic language is considered as one of the most widely used languages in the world. In fact, it is the native for about 330 million in the world. However, the current work on Arabic natural language processing is still limited due to several challenges. The main reasons for these challenges are: the rich morphology of Arabic language, its high degree of ambiguity, and Arabic dialects [2].

As a result; several Arabic tools have been implemented. Some of them are implemented for basic tasks of Arabic NLP: such as “MADAMIRA” [2], segmenters such as “FARASA” [3], libraries such as “AraNLP” [4] and “coreNLP”, or as an intermediary step for other NLP steps such as “ADAM” [5] which is used to solve other NLP problems such as automatic translation.

Another research discipline is to support textual analysis tasks by creating analysis tools. Large amount of data is now available on the web where a large number of text documents are loaded on a daily basis. Most of these documents are stored in an unstructured format, where the user has difficulty finding their needs. Therefore, there is an increasing need to automatically classify these documents based on their content into objective categories or classes to facilitate the retrieval of relevant documents [6].

Based on those challenges, we have created a program that helps in analyzing text. The created tool is similar to Posit text profiling toolset [7], which is a text-profiling tool for English. The approach aims at providing a general Arabic text profiling toolset that can be used in various corpus analysis projects. It focuses on three aspects of textual analysis: The first part is POS profiler; which performs the analysis on the corpus to derive statistics on the characteristics of (POS) in that corpus. The second is vocabulary profiler; it uses the output of POS profiler to determine the least common words in a given text; this will be helpful to know the main keywords of that text. The third is the readability profiler, which focuses on the ability to read text by evaluating a given document and giving it a score, which is a good indicator of the ambiguity level in the document and the readability of that document.

The rest of this paper is organized as follows: Section II discusses and compares the related research with the proposed one. Section III presents an overview of the created tool set. Section IV describes the experiments that have been conducted to evaluate the tool. Section V concludes the paper and discusses the scope of future work.

II. Related Work

With the rapid change in the forms and amount of data, it becomes difficult to analyse it without automated Natural Language processing techniques. Written text is available everywhere in the era of social media. There are other sources of textual data such books, magazines, newspapers emails and blog posts. Text-based content is necessary for effective communication. There have been several works on natural language processing for Arabic Language. Some of them focus on text categorization and classification. The author in [3] proposes Percentage and Difference Categorization (PDC) algorithm that categorizes the text taken from Arabic Wikipedia; it focuses on a hierarchy of main categories and subcategories of the text. The algorithm consists of two phases; in the first phase, it uses Basic Categorization Algorithm (BCA) to find the main category of the text. The second phase focuses on finding subcategory that the text

Manuscript received November 7, 2018; revised February 11, 2019.

Khalid M. O. Nahar and Malek Barahoush are with the Department of Computer Sciences, Faculty of IT and Computer Sciences, Yarmouk University, Irbid, 21163, Jordan (Corresponding author: Khalid M.O. Nahar; e-mail: khalids@yu.edu.jo)

F. Al Eroud and Abdallah M Al-Akhras are with the Department of Computer Information System, Faculty of IT and Computer Sciences, Yarmouk University, Irbid, 21163, Jordan.

doi: 10.18178/ijmlc.2019.9.2.790

belongs to.

Some researchers developed a framework for classifying Arabic dialects using probability models across social media data sets to categorize text into different Arabic dialect. The authors conduct a series of experiments using two different approaches; character n-gram Markov language model and Naïve Bayes classifiers [4]. Some researchers use n-grams of part-of-speech tags to determine whether they can be a distinguished when different categories are used. They use two classification methods, Naïve Bayes Classifier and Multinomial Naïve Bayes Classifier. Experiments were performed on five n-grams (n=1, 2, 3, 4, 5) lengths and two sets of tags (CLAWS5 tag set and simplified part-of-speech tag set). The results show a strong relationship between information about n-grams of part-of-speech tags and category of the text [5], [6].

Some works focus on categorizing e-mail content with a wide range of personal e-mail messages. The approach in [7] classifies emails dataset using two methods, the first depends on the WordNet class using support vector machine (SVM), and the second relies on clustering and classification- using K-Means algorithm.

The main challenge of building any NLP system for Arabic Language is the lack of language resources such as tagged corpora, tag set, and toolsets. POS tagging is not well studied in the Arabic language [8]. In fact, there is no standard POS tag set for Arabic Language Processing (ALP). Furthermore, it is hard to take advantages of existing POS taggers. Previous researchers propose tag sets that fit their research objectives without focusing on Arabic grammatical features. In addition, most researchers use tag sets derived from English. A new approach has been introduced for POS tagging of Arabic text. The authors suggested in [9] a criteria to design standards that could be used in the development of POS tagging for diverse types of text such as Classical Arabic and Modern Arabic Standard.

The authors in [10] proposed a new approach for POS tagging and lemmatization to solve a problem caused by the use of Hidden Markov Model (HHM). The latter had difficulty in estimating transition probability for small training corpus. They implemented POS tagger based on estimating transition probabilities using the decision tree approach. The result showed satisfactory accuracy with high-speed tagging process.

In [11], the authors proposed a new part-of-speech tag set category that was adapted to Arabic language. Instead of considering three standard classes of the tag set (Noun, verb, particle), the authors enlarged the tag set to seven classes. each class is subdivided into subclasses. This technique allows the Arabic terms to be categorized and then the most relevant morpho-syntactic feature for each word is extracted. Subclasses are extracted in new classes by applying a linguistic-based evaluation.

There are several other triggers-based approaches. One of those approaches has been introduced to create a text analysis tool [12]. The approach is based on the combination of high accuracy taggers “MADA”, “MXL” and “AMIRA”. This combination leads to a significant improvement in the overall accuracy of the proposed tool.

Other researchers have focused on implementing tools to address some of the essential tasks of NLP, such as morphological analysis, part-of-speech tagging, tokenization, lemmatization, discretization and named entity recognition. For instance, “ADAM” was designed to be used as a part of machine translation tasks. It has the advantage of short implementation time compared to analyzers that took years and required expensive resources [13]. Another morphological analyst is “MADAMIRA”, which presents a system for morphological analysis and disambiguation of texts. The system was implemented by combining many of the previous works such as “MADA” and “AMIRA”. In “MADA” (SVM) and N-gram language models are used to produce a list of every possible morphological explanation of each word “AMIRA” is based on supervised learning approach and it has been developed without any dependency on deep morphology. This combination introduces a quite efficient system [14]. Many other authors propose morphological analyzers.

TABLE I: SUMMARY TABLE OF RELATED WORKS

Group	Approach	Author and Year
Tools	MADAMIRA: a morphological analyser system for morphological analysis and disambiguation of texts	(A. Pasha *et al.*, 2014)
	FARASA: an Arabic segmenter based mainly on SVM-rank using liner kernel	(A. Abdelali *et al.*,2016)
	ARANLP: tools that have libraries for general Arabic NLP tasks.	(M. Althobaiti *et al.*, 2014)
	ADAM : analyzer used as a part of machine translation tasks	(W. Salloum and N. Habash, 2014)
	[15] Enhance the benchmark of Arabic morphological analyzers by the creation of the annotated corpus and presenting a new evaluation matric called GM-score.	(Y. Jaafar *et al.*, 2016)
	[19] Integrates the best tools of existing Arabic tools into a new toolkit.	(H. Rabiee, 2011)
	[20] a new tool for analyzing Arabic and English large texts. It provides corpus-linguistic analysis features.	(S. Almujaiwel and A. Al-Thubaity, 2016)
POS Tagger and tag- set	[9] Suggest criteria to design standard tagset that can be used in the progression of POS tagging.	(I. Zeroual *et al.*, 2017)
	[10] Implement POS tagger based on estimating transition probabilities using a decision tree approach.	(Z. Imad and L. Abdelhak, 2016)
	[11] Propose a new tagset adapted to Arabic language.	(Y. O. M. Elhadj *et al.*, 2014)
	[12] a New approach that combine taggers “MADA”, “MXL” and “AMIRA”.	(Alabbas.M and Ramsay.A)
Text Categorization	[3] Proposes Percentage and Difference Categorization (PDC) algorithm that categorizes text taken from Arabic Wikipedia.	(A. Yahya and A. Salhi)
Classification Text	[4] Develop a framework for Arabic dialects classification using probabilistic models across social media data sets.	(V. Bobicev *et al.*, 2014)
	[6] Use n-grams of POS tags to determine if it can be a discriminator of different. Classes. classification methods Naïve Bayes and Multinomial Naïve Bayes Classifiers are used for classification	(X. Tang and J. Cao)
	[7]Present and approach for email content classification. It based on word net using SVM and on clustering using k-means.	I. Alsmadi and I. Alhami, 2015)

The authors in [15] proposed an enhancement on the benchmark of Arabic morphological analyzer. Where an annotated corpus was created and proved by a linguistic expert. The corpus consists of 100 words from the holy Quran and each word in the corpus composes all possible morphological analyses. They also presented a new evaluation matric called GM-score, which takes into consideration the accuracy and execution time. The result is compared with three Arabic morphological analyzers BAMA, Alkhalil, and MADAMIRA. “Farasa” is an Arabic segmenter. The innovative aspect in “Farasa” is that it depends primarily on SVM-rank that uses liner kernel. The segmenter uses several properties and lexicons to evaluate the candidate segmentations of the word. The proposed approach was applied in two NLP tasks: machine translation and information retrieval [16], [17].

Another Arabic NLP tool for Non-Native Speakers is AraNLP”, which builds tools that have libraries for general NLP tasks. Some of these tools provide a Java library for Arabic text tasks. This tool introduces the feature that can be used without any compatibility issues. The tool includes tasks for sentence detector, tokenizer, light stemmer, root stemmer, part-of-speech tagger (POS-tagger), word segmenter, normalizer, punctuation, and diacritic remover [18]. New tools are integrated by combining the tool to achieve the best performances. The authors in [19] Compared the existing tools in terms of POS tagger and morphological analyzer. They integrate the best tools into a new toolkit. The metric of choice among tools is accuracy. The results showed that the best morphological analyzer is Alkhalil. In addition, the best POS tagger is Stanford. The newly integrated toolkit is tested in Modern Standard Arabic.

[20] proposed a new tool for analyzing large Arabic and English texts. The tool provides corpus-linguistic analysis features. The developed features include Chi-square, Log-likelihood, the Weirdness Coefficient WC, Mutual Information, Dice Coefficient and LogDice measure.

Table I summarizes the previous related works with some comparisons.

III. SAP Toolset Overview

SAP toolset concentrates on three related parts of the textual analysis. The first one is (POS) that performs an analysis on a given text to extract some statistical characteristics of that text. This module is known as POS Profiler.

The second is the Vocabulary Profiler. The output of the statistical data from (POS) Profiler is used by Vocabulary Profiler to determine the relative frequency of occurrences of vocabulary in the text. SAP Vocabulary Profiler is designed to allow users to compare the text with the Open Source Arabic Corpus (OSAC) for two news agencies (CNN and BBC) [19].

The third is the Readability Profiler. The Readability Profiler focuses on the results obtained from (POS) Profiler and Vocabulary Profiler to assess readability level for given document. Fig. 1 and Fig. 2 provide a general overview of the SAP tool. SAP toolset has many benefits. There are several tasks that can be performed by SAP textual analysis toolset, such as the generation of multi-word units and associated part-of-speech components. In addition, frequency analysis of the text can also be achieved. These features can be used as an initial step to classify web pages [20].

IV. Experiments and Evaluation

The following sections describe the functionalities and operations of the tool and are organized as follows: the nature and operations of the POS Profiler, the Vocabulary Profiler and the Readability Profiler.

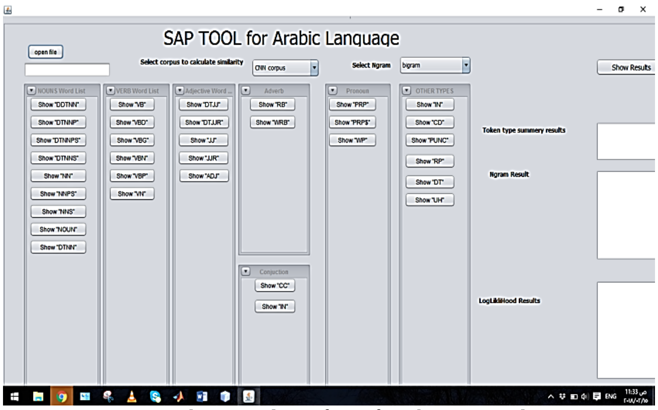

Fig. 1. The general interface of Arabic SAP tool.

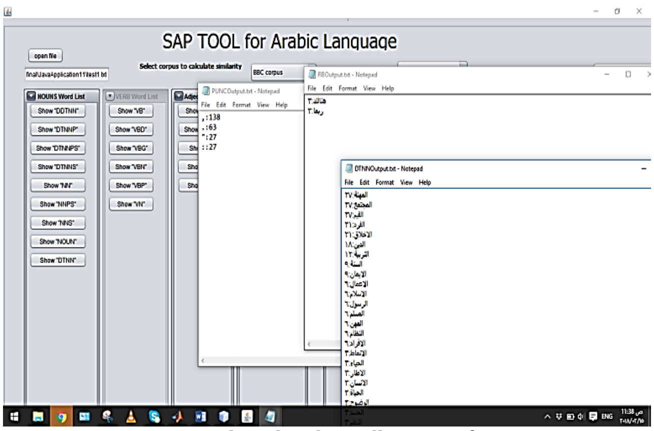

Fig. 2. SAP tool results when calling some functions.

A. *POS Profiler*

The POS profiler is designed work on part-of-speech profiling aspects of the text. The POS provides a detailed count of word occurrences for the text. It provides the user with a general statistic related to some part-of-speech as shown in Table II.

TABLE II: General POS Statistics

Total Words (tokens)
Total Unique words
Type/Token Ratio (TTR)
Number of sentences
Average Sentence Length (ASL)
Number of characters
Average word Length (AWL)

The output from the tool includes features such as the total words (tokens), total unique words (types), and type/token ratio, number of sentences, average sentence length, number of characters, and average word length. In addition, the total number for each token type and the tokens belong to this POS type is extracted. The forms of the Arabic language tags are defined according to Stanford (coreNLP) tag set. Furthermore, the total number of each POS type is determined. It is also defined as a set of part-of-speech that can be recorded by the profiler according to Stanford (coreNLP) tagger. Table III and Table IV show the Statistics of POS profiler, and the Token Types for each tag of Arabic part of speech based on Stanford tag set.

TABLE III: TOKEN TYPES BY POS BY [19]

Stanford Arabic POS	Tag set	abbreviation
Noun	noun, singular or mass with the determiner “Al” (ال)	DTNN
	Proper noun, singular with the determiner “Al” (ال)	DTNNP
	Proper noun, plural with the determiner “Al” (ال)	DTNNPS
	noun, plural with the determiner “Al” (ال)	DTNNS
	noun, singular or mass	NN
	Proper noun, singular	NNP
	Proper noun, plural	NNPS
	noun, plural	NNS
	noun	NOUN
Verb	verb, base form	VB
	Verb, past tense	VBD
	verb, gerund or present participle	VBG
	verb, past participle	VBN
	Verb, non-3rd person singular present	VBP
	verb, past participle	VN
Adjective	adjective with the determiner “Al” (ال)	DTJJ
	adjective, comparative with the determiner “Al” (ال)	DTJJR
	adjective	JJ
	Adjective, comparative	JJR
	Adj	ADj
Adverb	particle	RB
	Wh-adverb	WRB
Conjunction	Coordinating conjunction	CC
	Preposition or subordinating conjunction	IN
Preposition	Preposition or subordinating conjunction	IN
Pronoun	Personal pronoun	PRP
	Possessive pronoun	PRPS

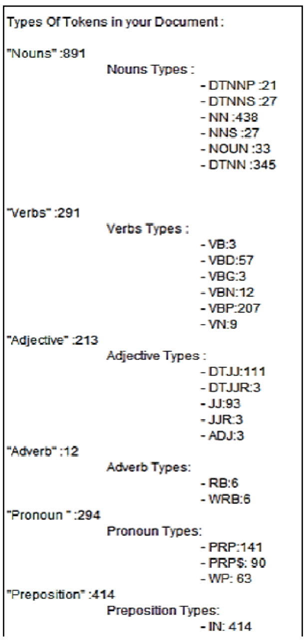

Fig. 3. Token types statistics (part 1).

Fig. 3 and Fig. 4 show POS profiler and the results generated by the SAP tool.

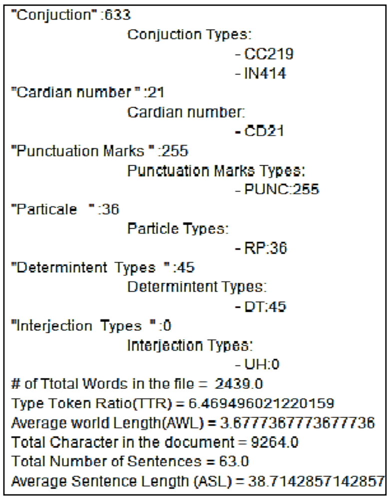

Fig. 4. Token types statistics (part 2).

A separate output file is provided for each of these POS types. Each file contains a list of words which belong to that POS type ordered by their frequencies. Another output file that is generated by POS Profiler and used by Vocabulary Profilers the top ten most frequent words. The file contains the top ten words and their frequency in a text. In addition, the POS Profiler finds the average sentence length and the total number of characters. These factors are used for the calculation of Flesch Reading Ease Formula [21].

B. *Vocabulary Profiler*

The SAP Vocabulary Profiler uses the results obtained by POS Profiler to find the most common words in the text according to the reference lists; CNN, BBC and a combination of both (OSAc). This step support finding the most common words in a text to determine the keywords of that text based on the Log-likelihood measure.

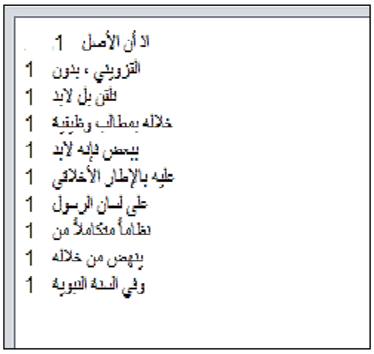

Fig. 5. Trigram sentences and frequencies.

We find the similarity between the text and the three reference lists using the Log-likelihood measure. We first retrieve the most frequent words in the text from the (POS) Profiler. In addition, the frequency in the three-reference list is extracted. The similarity between the text and the user choice is calculated by applying the log-likelihood measure[22], see equation 1.

$$L = 2 * \left(a * \log_{10}\left(\frac{a}{E1}\right) + b * \log_{10}\left(\frac{b}{E2}\right)\right) \quad (1)$$

$$\text{where } E1 = \frac{c*(a+b)}{(c+d)} \quad , E2 = \frac{d*(a+b)}{(c+d)} \quad (2)$$

where,

a= Frequency of word in the text
b=Frequency of word in the reference corpus
c=Total number of words in the text
d=Total number of words in reference corpus

In addition, SAP vocabulary analysis is expanded to consider n-gram frequencies within the analyzed text. N-gram frequency analysis allows you to choose the value of *n* in the n-gram. Three n-grams are used in the SAP tool: bigrams, trigrams and quad grams. Fig. shows an example of trigram results and their frequencies using the SAP tool.

C. Readability Profiler

This part of the proposed toolset focuses on the possibility of reading the text based on the statistical analysis generated using the POS profiler. Readability profiler measures the comprehensibility of a particular text. In particular, we are talking about the possibility of being understood by different readers with different educational level.

There are several readability metrics to assess documents. In our work, we used "Flesch Reading Ease Readability Formula" which is based on the average sentence length and the average number of syllables per words. It is a simple method to measure the grade-level of the reader. It is also one of the few accurate methods on which we can use without complex and inefficient calculations [23].

The value of Flesch Reading Ease Readability (RE) is given by equation 3:

$$RE = 206.835 - 1.015 * ASL - 84.6 * ASW \quad (3)$$

where,
ASL: is the ratio between the number of words and the number of sentences
ASW: is the ratio between the number of syllables and the number of words

Counting syllables in Arabic depends on its length [23]. It can be categorized to short, long or stress. Short syllables are either single constant or single constant plus short vowel (fatha, damma or kasra). On the other hand, longs are constantly followed by a long vowel (alef, waw or yaa'A). Stress syllables are tanween fatih, tanween damm, tanween kasr, and shadda. ASW is computed as shown in equation 4.

$$ASW = (2 * (\text{long} + \text{stress}) + \text{short})/\text{numberof words} \quad (4)$$

RE values is in the range 0 – 100. The higher number means that it is the easier to read. Table IV shows document assessment based on RE values. Fig. 6 demonstrates an example of applying SAP Readability module on a document.

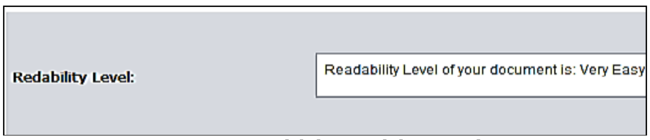

Fig. 6. SAP readability module example.

We submitted one hundred (100) text files to a full professor in Arabic Language as a human expert to judge their readability. Our aim was to compare the automated results done by computer through our tool with the assessment of human experts. The human expert gave a value from 0 to 100 for each text file based on the readability assessment measure in Table IV.

TABLE IV: READABILITY ASSESSMENT

Range	Textual evaluation
90 – 100	Very Easy
80 – 89	Easy
70 – 79	Fairly Easy
60 – 69	Standard
50 – 59	Fairy Difficult
30 – 49	Difficult
0 - 29	Very Difficult

TABLE V: READABILITY EVALUATION OF FILES 1-26

Readability Evaluation				
File Number	Size In KB	Human Expert Evaluation	Our Tool Evaluation	Hit/ Miss
1	363	79	75	1
2	152	49	57	1
3	96	65	57	1
4	115	78	71	1
5	234	61	60	1
6	175	82	77	1
7	281	35	48	0
8	297	39	35	1
9	124	30	29	1
10	219	86	78	1
11	6	58	63	1
12	170	70	75	1
13	310	59	51	1
14	113	61	52	1
15	191	5	11	1
16	288	70	72	1
17	36	100	100	1
18	83	68	63	1
19	52	66	45	0
20	395	79	72	1
21	292	25	16	1
22	66	55	46	1
23	115	45	40	1
24	75	99	92	1
25	241	91	79	0
26	298	68	68	1
27	223	42	33	1
28	194	9	13	1
29	36	52	55	1
30	112	78	71	1
31	325	74	69	1
32	343	97	99	1
33	250	63	55	1
34	369	5	22	0
35	350	48	16	0
36	14	69	76	1
37	111	42	94	0
11	6	58	63	1
12	170	70	75	1
13	310	59	51	1
14	113	61	52	1
15	191	5	11	1
16	288	70	72	1
17	36	100	100	1
18	83	68	63	1
19	52	66	45	0
20	395	79	72	1
21	292	25	16	1
22	66	55	46	1
23	115	45	40	1
24	75	99	92	1
25	241	91	79	0
26	298	68	68	1

TABLE VI: READABILITY EVALUATION OF FILES 27-53

Readability Evaluation				
File Number	Size In KB	Human Expert Evaluation	Our Tool Evaluation	Hit/Miss
27	223	42	33	1
28	194	9	13	1
29	36	52	55	1
30	112	78	71	1
31	325	74	69	1
32	343	97	99	1
33	250	63	55	1
34	369	5	22	0
35	350	48	16	0
36	14	69	76	1
37	111	42	94	0
38	339	63	65	1
39	249	93	95	1
40	361	55	61	1
41	61	64	63	1
42	63	88	51	0
43	356	3	7	1
44	199	88	75	0
45	389	90	84	1
46	68	90	93	1
47	210	75	69	1
48	56	50	81	0
49	386	10	18	1
50	146	57	63	1
51	331	21	83	0
52	167	61	66	1
27	223	42	33	1
28	194	9	13	1
29	36	52	55	1
30	112	78	71	1
31	325	74	69	1
32	343	97	99	1
33	250	63	55	1
34	369	5	22	0
35	350	48	16	0
36	14	69	76	1
37	111	42	94	0
38	339	63	65	1
39	249	93	95	1
40	361	55	61	1
41	61	64	63	1
42	63	88	51	0
43	356	3	7	1
44	199	88	75	0
45	389	90	84	1
46	68	90	93	1
47	210	75	69	1
48	56	50	81	0
49	386	10	18	1
50	146	57	63	1
51	331	21	83	0
52	167	61	66	1
53	307	17	26	1

TABLE VII: READABILITY EVALUATION OF FILES 54-79

Readability Evaluation				
File Number	Size In KB	Human Expert Evaluation	Our Tool Evaluation	Hit/Miss
54	97	22	13	1
55	247	70	74	1
56	124	52	50	1
57	127	22	29	1
58	394	70	77	1
59	296	12	56	0
60	280	98	96	1
61	329	77	71	1
62	4	6	93	0
63	388	40	31	1
64	21	27	36	1
65	241	100	85	0
66	277	40	47	1
67	73	21	40	0
68	115	88	87	1
69	329	34	29	1
70	214	77	34	0
71	284	95	96	1
72	35	55	62	1
73	164	90	93	1
74	149	66	69	1
75	223	90	86	1
76	171	73	81	1
77	132	31	82	0
78	211	76	78	1
79	93	36	29	1
80	299	91	95	1
54	97	22	13	1
55	247	70	74	1
56	124	52	50	1
57	127	22	29	1
58	394	70	77	1
59	296	12	56	0
60	280	98	96	1
61	329	77	71	1
62	4	6	93	0
63	388	40	31	1
64	21	27	36	1
65	241	100	85	0
66	277	40	47	1
67	73	21	40	0
68	115	88	87	1
69	329	34	29	1
70	214	77	34	0
71	284	95	96	1
72	35	55	62	1
73	164	90	93	1
74	149	66	69	1
75	223	90	86	1
76	171	73	81	1
77	132	31	82	0
78	211	76	78	1
79	93	36	29	1

TABLE VIII: READABILITY EVALUATION OF FILES 80-100

Readability Evaluation				
File Number	Size In KB	Human Expert Evaluation	Our Tool Evaluation	Hit/Miss
80	299	91	95	1
81	253	75	73	1
82	38	68	65	1
83	355	77	84	1
84	183	88	80	1
85	245	69	78	1
86	369	80	74	1
87	342	83	86	1
88	151	80	83	1
89	376	83	83	1
90	283	62	51	0
91	88	65	64	1
92	17	92	90	1
93	385	92	100	1
94	298	51	57	1
95	126	65	57	1
96	46	87	89	1
97	131	21	12	1
98	299	91	98	1
99	231	77	76	1
100	520	52	31	0
Accuracy of files 1 -100 = Number of Hits/100 = 81%				

In Table V, VI, VII, and VIII, the distance between Human and machine evaluation is calculated on the basis of Equation 5.

$$\textbf{Distance} = |\text{HumanEvaluation} - \text{MachineEvaluation}| \quad (5)$$

Then, if the distance between human expert and the tool is within 9 points, it is considered a hit for both machine and human. The number 9 indicates the length of interval between any two ranges in the assessment criteria. On the other hand, if the distance is greater than 9 it is considered a miss for the machine.

The accuracy measurement is calculated by counting the number of hits in Table V and then dividing the result by the total number of files as shown in equation 6.

$$\textbf{Accuracy} = \frac{\text{Number of Hits}}{\text{Number of Files}} * 100\% \qquad (6)$$

By applying the readability model on 100 Arabic text files and comparing them with the results obtained by human expert, an accuracy of 81% was obtained. A pictorial view of the results is shown in Fig. 7.

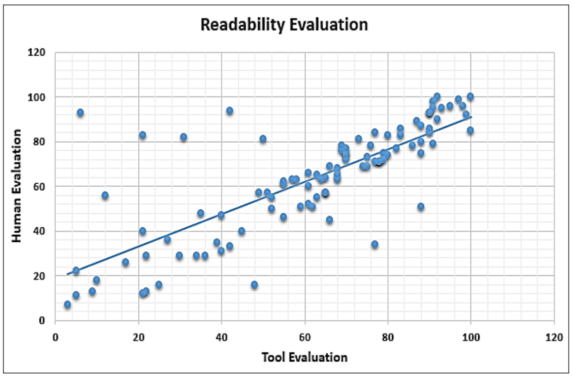

Fig. 7. Readability results graph.

The plan in Fig. 7 represents the complete match between human expert and machine evaluation. As seen from Fig. 7, most of the points are close to the hyper plan and some are on the hyper plan itself. Some points were far from the hyper plan; those represent 19% in our experiment.

V. SAP Toolset Contribution

SAP combines a variety of useful textual analysis facilities. The power of the SAP tool lies in its ability to manage arbitrarily large sizes of input, as well as their flexibility and extensibility. Current version of English toolset which is called Posit tools [7] relay on a Linux-based command line interface that users become acquainted with a range of commands to use the system effectively with no Arabic language support. In our research, we developed a convenient user-friendly graphical user interface which supports Arabic text processing facilities. In addition, this tool adds a useful feature, by inserting results in a database. Compared to classical Posit toolset [7], our tool adds the readability module for Arabic language.

Using SAP analysis toolset, Syntax frequency analysis, Multi-word units, associated POS-tagging, machine learning algorithms and knowledge extraction tools, we can create models to detect the terrorism-based context and suspend suspicious accounts that distribute counterfeit news. Therefore, SAP tool set that is developed along with above mentioned techniques are quite useful for classifying the web contents including good and suspicious contents.

VI. Conclusion

In this paper, we created a SAP Arabic text profiling toolset. It consists of three modules working together to provide some text analysis facilities: POS, vocabulary profiler and readability profiler. The POS uses Stanford coreNLP tagger to accomplish the POS profiler statistics. In addition, the Vocabulary profiler provides the user with the statistical results needed to aid authors who want feedback on their vocabulary usage. In the current form, SAP toolset provides the user with an easy to use graphical user interface. In addition, it allows the user to compare his/her text with OSAC corpus in term of vocabulary frequency. Readability profiler assesses a given document using Flesch Reading Ease Readability Formula which is a good indicator of the ambiguity of a given text. The readability accuracy of the tool has been measured by comparing it to human experts in one hundred text files. The readability tool accuracy reaches 81%.

In the future, we aim to enhance the running time of SAP in order to be able to compare large corpora in less time. Internet web sites that contain terrorism related contents are considered one of the main factors for radicalization among young adults. Due to these web sites, youth may contribute to terrorist activities. Collecting vast amounts of terrorism and extremism data by retrieving the web-pages visited is our future extension to this work to stop possible terrorist acts. A knowledge extraction can be deployed on the results using SAP analysis toolset. This leads to automate the evaluation of IR systems by creating a matching between manual and automatic classification. Using techniques such as, SAP analysis toolset, Syntax frequency analysis, Multi-word units, associated POS-tagging, machine learning algorithms and knowledge extraction tools, we can create models to detect the terrorism-based context and suspend suspicious accounts that distribute fake news. Therefore, SAP tool set will be quite useful for classifying the web contents including good and suspicious contents.

Acknowledgement

We extend our sincere thanks to Yarmouk University - Irbid - Jordan, where this scientific research was supported by the Deanship of Scientific Research and Graduate Studies at the University.

References

[1] K. M. O. Nahar, N. Alhindawi, O. M. Al-hazaimeh *et al.*, *NLP and IR Based Solution for Confirming Classification of Research Papers*, vol. 96, no. 16, pp. 5269–5279, 2018.

[2] K. Nahar, H. Al-Muhtaseb, W. Al-Khatib, M. Elshafei, and M. Alghamdi, "Arabic phonemes transcription using data driven approach," *Int. Arab J. Inf. Technol.*, vol. 12, no. 3, pp. 237–245, 2015.

[3] A. Yahya and A. Salhi, "Arabic text categorization based on Arabic Wikipedia," *ACM Trans. Asian Lang. Inf. Process.*, vol. 13, no. 1, pp. 1–20, 2014.

[4] V. Bobicev, M. Sokolova, and M. Oakes, "Recognition of sentiment sequences in online discussions," *Soc. NLP2014*, pp. 44–49, 2014.

[5] K. M. Nahar, "Off-line arabic hand-writing recognition using artificial neural network with genetics algorithm," *Int. Arab J. Inf. Technol.*, vol. 6, no. 6, 2018.

[6] X. Tang and J. Cao, "Automatic Genre Classification via N-grams of Part-of-Speech Tags," *Procedia - Soc. Behav. Sci.*, vol. 198, pp. 474–478, 2015.

[7] I. Alsmadi and I. Alhami, "Clustering and classification of email contents," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 27, no. 1, pp. 46–57, 2015.

[8] K. M. O. Nahar, M. Elshafei, W. G. Al-khatib, H. Al-muhtaseb, and M. M. Alghamdi, "Statistical analysis of arabic phonemes for Continuous

arabic speech recognition," *Int. J. Comput. Inf. Technol.*, vol. 1, no. 2, pp. 49–61, 2012.
[9] I. Zeroual, A. Lakhouaja, and R. Belahbib, "Towards a standard Part of Speech tagset for the Arabic language," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 29, no. 2, pp. 171–178, 2017.
[10] Z. Imad and L. Abdelhak, "Adapting a decision tree based tagger for Arabic," in *Proc. 2016 Int. Conf. Inf. Technol. Organ. Dev. IT4OD*, 2016, pp. 2–7.
[11] Y. O. M. Elhadj, A. Abdelali, R. Bouziane, and A. H. Ammar, "Revisiting Arabic Part of Speech Tagsets," in *Proc. IEEE/ACS Int. Conf. Comput. Syst. Appl. AICCSA*, 2014, vol. 2014, pp. 793–802.
[12] M. Alabbas and A. Ramsay, "Combining strategies for tagging and parsing Arabic," *Anlp 2014*, pp. 73–77, 2014.
[13] W. Salloum and N. Habash, "ADAM: Analyzer for Dialectal Arabic Morphology," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 26, no. 4, pp. 372–378, 2014.
[14] M.-B. ArfathPasha, "MADAMIRA : A Fast , Comprehensive Tool for Morphological Analysis and Disambiguation of Arabic," in *Proc. 9th Lang. Resour. Eval. Conf.*, 2014, pp. 1094–1101.
[15] Y. Jaafar, K. Bouzoubaa, A. Yousfi, R. Tajmout, and H. Khamar, "Improving Arabic morphological analyzers benchmark," *Int. J. Speech Technol.*, vol. 19, no. 2, pp. 259–267, 2016.
[16] M. N. Al-Kabi, G. Kanaan, R. Al-Shalabi, M. O. K. Nahar, and M. B. Bani-Ismail, "Statistical classifier of the Holy Quran Verses (Fatiha and Yaseen Chapters)," *J. Appl. Sci.*, vol. 5, no. 3, pp. 580–583, 2005.
[17] A. Abdelali, K. Darwish, N. Durrani, and H. Mubarak, "Farasa: A Fast and Furious Segmenter for Arabic," in *Proc. 2016 Conf. North Am. Chapter Assoc. Comput. Linguist. Demonstr.*, 2016, pp. 11–16.
[18] M. Althobaiti, U. Kruschwitz, and M. Poesio, "AraNLP: a Java-based Library for the Processing of Arabic Text.," in *Proc. Ninth Int. Conf. Lang. Resour. Eval.*, pp. 4134–4138, 2014.
[19] H. Rabiee, *Arabic Language Analysis Toolkit*, 2011.
[20] S. Almujaiwel and A. Al-Thubaity, *Arabic Corpus Processing Tools for Corpus Linguistics and Language Teaching*, vol. 2, August 2016.
[21] J. P. Kincaid, R. P. Fishburne, R. L. Rogers, and B. S. Chissom, *Derivation of New Readability Formulas (Automated Readability Index, Fog Count and Flesch Reading Ease Formula) for Navy Enlisted Personnel*, 1975.
[22] P. Rayson, D. Berridge, and B. Francis, "Extending the Cochran rule for the comparison of word frequencies between corpora," *JADT 2004 7es Journées Int. d'Analyse Stat. des Données Textuelles*, pp. 1–12, 2004.
[23] M. El-haj and P. Rayson, "OSMAN – A novel arabic readability metric," in *Proc. Lang. Resour. Eval. Conf.*, 2016, pp. 250–255.

Khalid M. O. Nahar received his BS, and MS degrees in computer sciences from Yarmouk University in 1992 and 2005 respectively. He was awarded a full scholarship to continue his PhD in computer science and engineering from King Fahd University of Petroleum and Minerals (KFUPM), KSA. In 2013 he completed his PhD and started his job as an assistant Prof. at Tabuk University, from 2013 to 2015. In 2015, he back to work at Yarmouk University-Jordan, as an assistant professor in the Department of Computer Science. For now, he is the dean assistant for quality control. His research interests include continuous speech recognition, Arabic computing, natural language processing, wireless sensor networks (WSNs), multimedia computing, content-based retrieval, artificial intelligence, and software engineering.

Ahmed Al Eroud is an assistant professor of computer information systems at Yarmouk University in Jordan. He holds degrees in information systems (ph.D. and M.S.) from the University of Maryland, Baltimore County, and Software engineering (b.s.) from Hashemite University in Jordan. He was a visiting associate research scientist at the University of Maryland, Baltimore County working on cyber security research projects. His research work focuses on cyber-security, data mining for privacy preserving network data analytics, and detection of social engineering attacks.

Malek Barhoush is an assistant professor in the Computer Science Department at Yarmouk University, Jordan. He received his Ph.D. in computer science from the Concordia University, Canada in 2012. He granted a scholarship for his master's degree at Yarmouk University (YU), he also granted a scholarship for his Ph.D. During his career at YU, he served as an assistant dean for Laboratory Affairs for information technology and computer science faculty, and he served as a chairman for network and information security department. He has several journals and a conference research publication in a number of research areas. His research interest focuses on computer network security, mobile computing, web intelligence, mobile sensors, parallel & distributed systems, and cloud computing.

Abdallah Al-Akhras is an instructor in the Department of Computer Information System, Faculty of IT, Yarmouk university, Irbid-Jordan. He received his BSc in computer science from Yarmouk University in Jordan at 2003 and he received his MSc in computer information system from the same university at 2005. From 2005 until now he works at Yarmouk University. His research interests include: information retrieval, natural language processing, data mining, artificial intelligence.