RESEARCH ARTICLE OPEN ACCESS

A New Approach to Romanize Arabic Words

*Hussein K. Khafaji, **Nada Adnan Taher
*Al-Rafidain University College/ Computer Communications Engineering Dept.
**Al-Rafidain University College Computer Engineering Dept.

ABSTRACT
Romanization of Arabic words has been acquired the interest of the researchers due to its importance in many fields such as security and terrorism fighting, translation, religious purposes, etc.
In this paper, a proposed method was presented to solve the drawbacks of available methods such as lack of reverse recognition, using of extra letters and punctuation characters, and neglecting the correlation of the letters in a word.
This method was implemented and tested using a sample of 100 undergraduate Iraqi students and 150 Arabic words which romanized using five well-known methods in addition to the proposed one. The test showed that the proposed method dominants the rest method from the recognition and reverse recognition process in considerable ratio.

I. INTRODUCTION

Sometimes, it is impossible to translate some of the words and terms from one language to another due to the nature of a language itself or unavailability of equivalent meaning word especially in the case of names of peoples, places, countries, and local names of things, ideas, and doctrines. Therefore, the need for Romanization or transliteration was emerging [1]. Transliteration is the conversion of a text from one script to another. If the language characters are Roman character, this process is called Romanization [1]. Romanization or latinization, in linguistics, is the conversion of writing from a different writing system to the Roman (Latin) script, or a system for doing so. Methods of Romanization include transliteration, for representing written text, and transcription, for representing the spoken word, and combinations of both [2]. Transcription methods can be subdivided into phonemic transcription, which records the phonemes or units of semantic meaning in speech, and more strict phonetic transcription, which records speech sounds with precision [3].

Most Arabic Romanization systems, which began by Orientals many centuries ago when the Arab Islamic civilization contacted with European civilization, are self-assiduousness and vary from one person to another. Romanization is often termed "transliteration", but this is not technically correct. Transliteration is the direct representation of foreign letters using Latin symbols, while most systems for Romanizing Arabic are actually transcription systems, which represent the sound of the language [4]. As an example, the rendering munāẓarat al-ḥurūf al-ʿarabiyyah of the Arabic: مناظرة الحروف العربية is a transcription, indicating the pronunciation; an example of transliteration would be mnaẓrẖalḥrwfalʿrbyẖ [5]. Another case, which differentiates between normalization and transliteration, is caused by the fact that written Arabic is normally unvocalized, i.e., many of the vowels are not written out, and must be supplied by the language [6]. Hence, unvocalized Arabic writing does not give a reader unfamiliar with the language sufficient information for accurate pronunciation. As a result, a pure transliteration, e.g. rendering "قطر" as qṭr, is meaningless to an untrained reader. For this reason, transcriptions are generally used that add vowels, e.g. qaṭar [5].

II. ROMANIZATION SYSTEMS

The first attempts of Romanization began by Orient lists based on personal reasoning in how to write and pronounce the Arabic word in Latin characters. This matter has several problems, including that the Arabic word can be romanized in a number of ways depending on the different place or origin, even if all of them took into account the eloquent speech as much as possible [7]. Add to this that the process of Romanization affected by the mother tongue of Orientals. English may use letter 'a' to denote the slot, sh denotes the character s, while he may use his German counterpart, crafts e to denote the slot, sch or ch to denote the letter s, etc. Therefore, many approaches were emerging to Romanize Arabic words [8].

\Currently, the Arab linguists, orient lists, and offices follow manual process of Romanization in most cases, which result in the followings:

1. Lack of consistency in the process of Romanization inside the text. For example, the Lawrence of Arabia in his famous book he wrote the same Arabic name several different ways, for

example, the word "Jeddah" and sometimes Jidda (the name of the famous city in the Kingdom of Saudi Arabia) as he wrote the name of Abdul - Almoeen six different ways. [9]

2. Lack of consistency in the process of Romanization between various texts, whether for the same person or a different group of people. This is a fortiori, of course. The surveyed write the name of Sharaf al-Din in many Romanized forms such as:
   SharafEldin, Sharaf El Din, SharafElDin, Sharaf-Eldin, Sharaf-El-Din, Sharafeldin, Sharafel Din, Sharafelddin, Sharafelldin, Sharafudin, Sharafuddin, Sharaf Aldine, Sharaf Al Din, Sharaf Al Dine, Sharaf-Al-Din, Sharaf El deen, etc[10].
3. Failure in the reverse process of Romanization, i.e., recovery of the Arabic word of the Romanized word.[11]
4. Slow "Romanization" process.
5. The possibility of human error due to the manual work [12].

Perhaps the logical solution to this is to develop a uniform system for the binding process of Romanization that means made standard specification for it. The exclamation point when you know that there are several specifications proposed and some are in circulation among the researchers, however, none of them has achieved sufficient deployment [13]. For example, there are currently on the scene several ways of Romanization, and perhaps surprising that many researchers, governments, and countries do not use any of these, but each may invent a way of Romanization. The followings are the most known Romanization approaches [14]:

1 - US Library of Congress (LC) and perhaps the most common.

2 - Specification of Romanization British BS4280 (BSI).

3 - Specification for the Department of Islamic knowledge.

4 - Specification for ISO.

5 - Specification of the international magazine for Middle East Studies (IJMES).

6 - Specification of the Institute of Islamic Studies, McGill University, Canada.

One of the disadvantages of the previous Romanization systems is the need to add special symbols for regular letters of the alphabet. This in turn leads to the difficulty of computing due to the lack of these symbols on the keyboard directly (such as putting points higher character that is not already exist)[14]. In addition, some of the Arabic letters are romanized by more than one Latin character, which may be signed in confusion. Moreover, the difficulty of remembering these symbols in general. Furthermore, the reverse process to Romanized name may not produce the original Arabic name [15]. Due to the mentioned drawbacks of the Romanization systems, the numerous scripting of Arabic names still a big problem in documents, passports, hospital records, terrorist tracing by Interpol, Academic certificates, etc. This leads to various problems, especially when retrieving the Romanized names from a database. Although the British Standard BS4280 is considered a reasonably consistent, but they do not use on a large scale. In the research [8] there is a comparison between these systems can be referenced. In spite of the presence of Standard Audio International Phonetic Alphabet highly complex but it fits specialists in the field of linguistics only [8]. It may seem to be a natural alternative to this is to compensate for each character Arabic counterpart voice of English (Roman) , but this is not available because most of the Arabic letters cannot be compensated by one character and that such symbols S. , for example, as to some of the letters is unrivaled audio in the target language [16].

III. REQUIREMENT FOR GOOD ROMANIZATION SYSTEMS

The following basic characteristics of the requirements necessary for the Romanization systems can be concluded:

1. Symmetry between each character in the source language (Arabic), the target language (English) and this condition is difficult to fully abide by it to the following:

A) Arabic characters need to diacritical marks so they can be tuned and writes these signs above or below the original character. Thus, the symmetry can be automatically bound by it (as the formation is independent characters) cannot be bound by it manually (where the accent character is only one character). [17]

B) In most cases cannot pronounce the name in true way for experienced people only [18].

2. The possibility of recovery of the Arabic script, which already Romanized to its original state without confusion [19].

3. Totalitarian in the sense that all the symbols and sounds of language in the source language have compared to the target language [20].

The aim of this research is design and implementation of a Romanization of Arabic words to avoid the mentioned drawbacks and support the requirements of robust Romanization systems.

IV. The Proposed Romanization System

The proposed Romanization system consists of the following components: Word's string handler, Irregular-names database scanner, Word scanner, and Romanizer, as shown in figure (1). These modules will be explained in the next sub sections.

The input to the RS is an Arabic word. The word should be written with all its HARACAT such as DAMMA, FATHA, SUKUN, etc. For example, the pronunciation of "روح" and "رَوْح" is different in spite of the similarity of the Arabic letters in these two words due to the difference of the HARACAT. In addition, the word may be compound or simple word, such as "صَلاحُ الدين", "شَجَرَةُ الدُّر", or "عَلِيْ". Also, the input word may be irregular word. Usually the number of irregular words is very small in comparison with the number of the language words. A special database was constructed to hold the Romanization of the irregular words. The database is scanned to find the Romanized word otherwise; the control will be transferred to word scanner and the Romanizer to construct the English word.

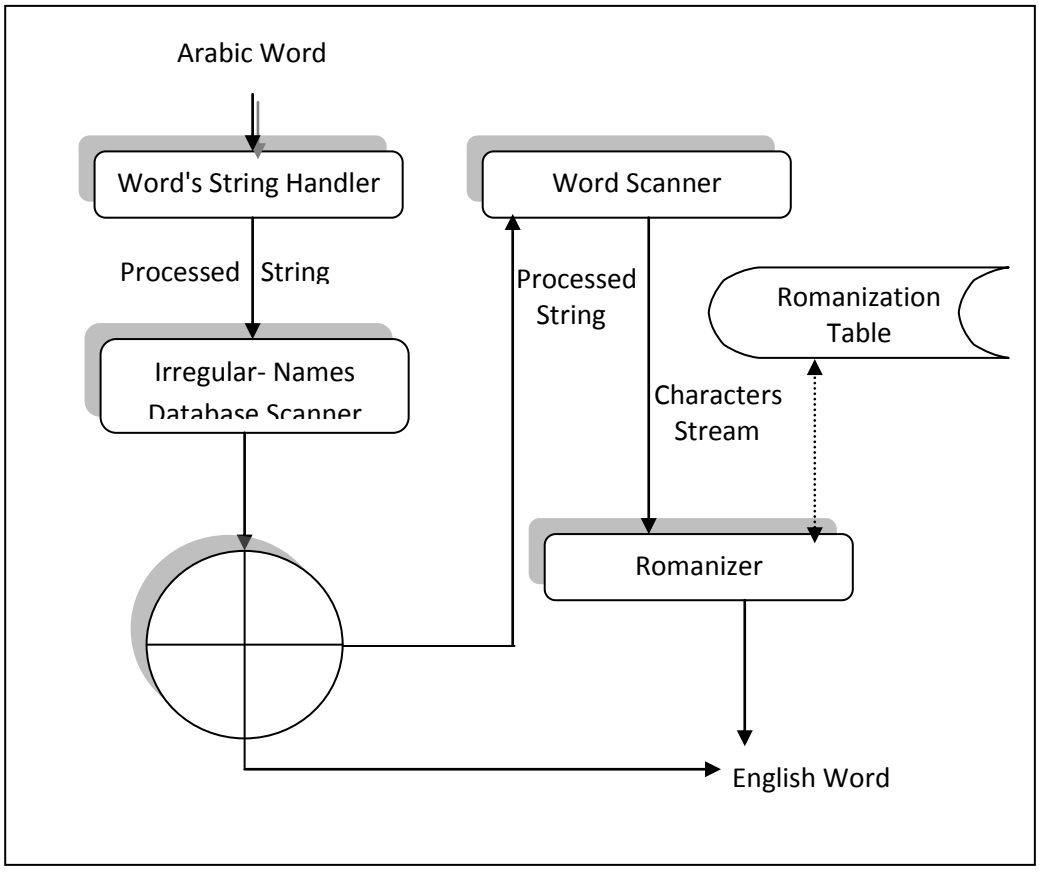

Figure (1) the proposed Romanization System Architecture

1. Word's String Handler

This module removes extra white spaces, punctuation, numbers, and special characters. After this trimming process the word will be checked by a spell checker designed according to [21].

Indeed, this module disintegrates the word to its characters and then reconstructs a word from the remaining letters because the stored words in the irregular-name database are stored as string data type. Figure (2) shows a self-documented pseudo code representing the duties of this module.

2. Irregular-name Database Scanner

As it is mentioned in section 2, there is a correspondence between the Arabic word writing and its pronunciation except some irregular name and word. For example, the words "الله" and "الرحْمٰن" are regarded as irregular words. Such these words, which are not obeyed to the writing rules, were stored in the irregular-name database. The input to this module is a string processed by the previous module, word's string handler, stored in out-string variable presented in figure (2). The output of this module is a string of zero or more characters. Empty string indicates unavailability of the word in the database, i.e., it is not 'irregular' word; therefore the control will be signaled to Romanization process.

```
...
Input-string=right trim (left trim (input-string));
While not end of input-string do
{
        ch=getcharacter(input-string);
        out-string="";
        Case ch of
        {
                ' ': { out-string+=' ';
                       skip-spaces(input-string);
                     }
                '-' : skip_TATWEEL_character(input-string);
                '0'..'9': skip_numbers(input-string);
                'ء'..'ي' : out-string+= ch;
                Default: skip_others(input-string);
        }
        out-string=spell-checker(out-string);
}
...
```

Figure (2) Pseudo code of word's string handler module

3. Word Scanner

This module will be activated when the word is regular word. It receives the processed word obtained from the word's string handler module and converts it to stream of letters and HARACAT to enable the next module; romanizer, to keep track of some pointers because the pronunciation of a letter may depend on some letters or HARACATs.

Arabic letters are divided into two classes; "Ashshamsia" and "Al-Qameria". Usually the words are prefixed with "Al"; "الـ". When a letter from "Ashshamsia" class is joined with "الـ", the "الـ" is not articulated but the letter is repeated. For example, the word "الروح" is written as "Arroh" and not "Al-roh". But "الـ" should be articulated with "Al-Qameria" letters such as "الكَنْز", "Al-Kanz". The output of this module according to first example is the stream ['ا', 'ر', 'رّ','و','ح'] and its output for the second example is the stream ['ا','ل','ك','َ','ن','ْ','ز']. One of the duties of this module is recognition of these two cases. Figure (3) shows a pseudo code of this process.

The Shamsyan function invoked in the pseudo code of figure (2) is a binary function returns true if the letter is from "Ashshamsia" class otherwise it returns false, i.e., it returns false if the letter is one of the following letters: هـ، م، ي، ق، ع، ف، خ، و، ك، ج، ح، غ، ب، أ .

```
...
ch=getcharacter(input-string);
Case ch of
'ا': if (ch=getcharacter(input-string)= ='ل' )
      {  ch=getcharacter(input-string)
         If   (shamsyan(ch)) initialize the stream with ['ا', ch, 'ّ'];
         Else initialize the stream with ['ا','ل',ch]
       }
    Else initialize the stream with ['ا', ch];
...
```

Figure (3) Pseudo code of recognition of "Ashshamsia" and "Al-Qameria" letters

4. Romanizer Module

The input of this module is the stream produced from the word scanner module. It depends mainly on the Romanization table presented in table (1).

Table (1) Romanization Table

No.	Unicode	Pronunciation	English Letter	Arabic Letter or HARACAT
1	0621		No thing	"Hamza" (ء)
2	0623	Like A in Apple	A	أ
3	0625		I	إ
4	0622		A	آ
5	0624	Like o in On	U	ؤ
6	FE8C		I	ـئـ
7	0627		A	ا
8	0628	Like o in Baby	B	الباء (ب)
9	062A	Like T in Tree	T	التاء (ت)
10	062B	Like the Th in Theory	Th	الثاء (ث)
11	062C	Sometimes like the G in Girl or like the J in Jar	J	الجيم (ج)
12	062D	Like the h in he yet light in pronunciation	H	الحاء (ح)
13	062E	Like the Ch in the name Bach	Kh	الخاء (خ)
14	062F	Like the D in Dad	D	الدال (د)
15	0630	Like the Th in The	Dh	الذال (ذ)
16	0631	Like the R in Ram	R	الراء (ر)
17	0632	Like the Z in zoo	Z	الزين (ز)
18	0633	Like the S in See	S	السين (س)
19	0634	Like the Sh in She	Sh	الشين (ش)
20	0635	Like the S in Sad yet heavy in pronunciation	S	الصاد (ص)
21	0636	Like the D in Dead yet heavy in pronunciation	D	الضاد (ض)
22	0637	Like the T in Table yet heavy in pronunciation	T	الطاء (ط)
23	0638	Like the Z in Zorro yet heavy in pronunciation	Z	الظاء (ظ)
24	0639	Has no real equivalent sometimes they replace its sound with the A sound like for example the name Ali for ع/ علي/ali	A	العين (ع)
25	0639+0650		E	العين (عـ): (عِ) or (عِـ)
26	0639+064E		A	العين (عـ): (عَ)
27	0639+0627		Aa	العين(عـ): (عا)
28	0639+064F		O	العين (عـ): (عُ)
29	0639+0627 or 0639+064E		A	العين (ـعـ): (ـعا) or(ـعَ)
30	0639+0647		Nothing	العين (ـعـ): (ـعُ) (ـعو)
31	064E+0639			العين (ـعـ): (ـَعْ)
32	063A	Like the Gh in Ghandi	Gh	الغين (غ)
33	0641	Like the F in Fool	F	الفاء (ف)
34	0642	Like the Q in Queen yet heavy velar sound in pronunciation	Q	القاف (ق)
35	0643	Like the K in Kate	K	الكاف (ك)
36	0644	Like the L in Love	L	اللام (ل)
37	0645	Like the M in Moon	M	الميم (م)
38	0646	Like the N in Noon	N	النون (ن)
39	0647	Like the H in He	H	الهاء (هـ)
40	064F+0648	Like U in soon	U	الواو (ساكنة وما قبلها مضموم): (ـُوْ)
41	0648	Like the W in the reaction of astonishment saying:	W	الواو (و)

		WAW!		
42	0650+064A		I	الياء (ساكنة وما قبلها مكسور): (ـِيْـ)
43	064A	Like the Y in you	Y	الياء (ي)
44	0649		A	الألف المقصورة (ى)
45	0629		H	التاء المربوطة الساكنة (ـةْ)
46	064F		U	الضمة (ـُ)
48	064C		Un	التنوين المضموم (ـٌ)
49	064E		A	الفتحة (ـَ)
50	064B		An	التنوين المفتوح (ـً)
51	0650		I	الكسرة (ـِ)
52	064D		In	التنوين المكسور (ـٍ)
53	0651		Duplicate the letter before SHADDA	الحرف المشدد (ـّ)

In many cases, the romanizer maintains pointers to the current letter and the previous three letters and HARACATs and this is the reason of constructing the output of the scanner as a stream to achieve fast movements for the romanizer over the word.

5. Experimental Results

A sample of 100 UG Arabic students was selected to read and recognize 150 Romanized Arabic words, and then write the corresponding Arabic words. Each word is Romanized by IPA, UNGEGN, ISO, Arabtex, ALA-LC, in addition to the suggested method, i.e., each student guessed 900 Romanized word presented in a random list. Fifty words of the 150 are frequent words. The rest 100 words are classified as relatively frequent and rarely used words because the aim of the experiment is measuring the ability of the persons to read and recognize the original word and prevent him from guessing the words. Table (2) shows the ratio of recognized words according to each approach.

Table (2) Recognition ratio of Romanized words

IPA	UNGEGN	ISO	Arabtex	ALA-LC	Suggested Method
%60	**%77**	**%79**	**%83**	**%69**	**%96**

All the unrecognized words in the suggested method are containing the letter (عـ), Ain of (Unicode=0639) with the letter Hamza, (Unicode=0621) or Alef, (unicod= 0623). For example the word "عَلي"-"Ali" and "عالي"-"Aali" are recognized as same word "عَلي"-"Ali" by nine students, in addition to the words "مَسْعول"-"Masool" and "مَسْؤول"-"Masaol" are recognized as "مَسْؤول"-"Masaol" by twelve student. The reason for that, according to our opinion, is the frequent use of the word "عَلي" and "مَسْؤول" in comparison with "عالي" and "مَسْعول". There are twelve students did not recognize the word "بَعْلَبَك". There are 86 students recognized all the words Romanized by the suggested method while table (3) presents the number of students who recognize all the words in the rest approach.

Table (3) Number of students who recognized all the words for each approach

IPA	UNGEGN	ISO	Arabtex	ALA-LC	Suggested Method
45	**56**	**62**	**69**	**71**	**86**

The average of recognition of the 100 students for each approach is presented in table (4) to achieve reliable results.

Table (4) Recognition average for each approach

IPA	UNGEGN	ISO	Arabtex	ALA-LC	Suggested Method
85.25	**84.20**	**82.50**	**83**	**89.33**	**94.45**

Consider table(3) and table (4); in spite the fact that there are 45 students recognized all the words Romanized by using IPA, the average of recognition is higher than UNGEN, ISO, and Arabtex; also it is very close to ALA-LC recognition average.

V. Discussion and Conclusions

This work presents a new approach to Romanize Arabic words. This approach characterized by many characteristics such as:

1. This approach uses English letters only for Romanized word without any sound, punctuation, or special characters; therefore it has got high readability according to the results presented in table (2).
2. It doesn't depend on letter to letter correspondence, but it considers the correlation of a letter with former letters and HARACATs, this makes the Romanization very close to the actual Arabic pronunciation. This feature simplifies the matching of Arabic word and its Romanization in case of availability or absence of the Arabic words.
3. It is the first approach which considers the properties of the letters such as "Ashshamsia" and "Al-Qameria" features to simulate the actual Arabic word articulate.
4. According to the experiment results, the students' incorrect recognition in the suggested approach occurred in the words containing the letters "أ" and "ع", but in the rest approaches most reading mistakes occurred in words containing the letters "أ", "ع", "ذ", "د", "ض", "ظ", and "ث".

From the implementation point of view, the preprocessing of the input string eases the Romanization operation and makes it very close to Arabic word pronunciation.

Bibliography:

[1] Toivonen, Jarmo, Ari Pirkola, Heikkikeshustalo, Kari Visala, and Kalervo Jarvelin, Translating cross lingual spelling variants using transformation rules", Information Processing and Management, 2004

[2] Meng, H.M.; Wai:-Kitlo; Berlin Chen; Tang, K. Automatic Speech Recognition and Understanding. *ASRU'01. IEEE work shop on*, 9-13 Dec. 2001. 311-214. 2001.

[3] UNGEGN working Group on Romanization Systems, Report on the UNITED NATIONS ROMANIZATION SYSTEMS CURRENT STATUS OF FOR GEOGRAPHICAL NAMES, Arabic version 2.2, January 2003.

[4]. Alghamdi, Mansour (2004) Analysis, Synthesis and Perception of Voicing in Arabic. Al-Toubah Bookshop, Riyadh. (The book is now available on the web)

[5]. Nahar, Khalid, Moustafa Elshafei, Wasfi Al-Khatib, Husni Al-Muhtaseb, Mansour Alghamdi, "Statistical Analysis of Arabic Phonemes for Continuous Arabic Speech Recognition", International Journal of Computer and Information Technology, 2012.

[6]. http://www.alab.com/arab/language/roman1. htm#TRANSCRIPTION a good discussion about the "standards" used for Romanization of the Arabic text.

[7]. "Wikipedia: Manual of Style/Arabic". Retrieved 2013-06-03.

[8]. ArabEasy - transliterate Arabic webpages

[9]. http:// transliteration.eki.ee/pdf/Arabic_2.2.pdf

[10]. Sharaf Eldin, A., 6. Hanna, S. and N. Greis, "Writing Arabic - A linguistic approach from sounds to script", E.J. Brill, Leiden, 1972. "Computer-Assisted Arabic Transliteration", 7th.. ICAIA, 1999.

[11]. Roochnik, P., "Computer-based solutions to certain linguistic problems arising from the Romanization of Arabic names", Ph.D. dissertation, Georgetown university, Washington DC, 1993.

[12]. Al-Onaizan, K. Knight, "Transliteration of Names in Arabic Text", obtained via http://www.isi.edu/natural-language/projects/rewrite/yaser-acl02-WS.ps

[13]. Arbabi, M., S. Fischthal,V. Cheng and E. Bort, "Algorithms for Arabic name transliteration", IBM J. of research and development, Vol. 38, no. 2, Mar., 1994, pp. 183-193.

[14]. "Arabic" (.pdf). *ALA-LC Romanization Tables*. Library of Congress. p. 9. Retrieved 2013-06-14.

[15]. Stalls, B. and K. Knight, "Translating Names and Technical Terms in Arabic Text", Proceedings of the COLING/ACL Workshop on Computational Approaches to Semitic Languages, 1998.

[16]. Beesley, K. R. , "Arabic Finite-state Morphological Analysis and Generation", In COLING-96 proceedings, volume 1 , pages 89-94 Copenhagen, Center for Sprogteknologi. The 16th International Conference on Computational Linguistics, 1996.

[17]. Arbabi, M., S. Fischthal, V. Cheng and E. Bort, "Algorithms for Arabic name transliteration", IBM J. of research and development, Vol. 38, no. 2, Mar., 1994, pp. 183-193.
[18]. Beesley, K. R., "Computer Analysis of Arabic Morphology: A two-level Approach with Detours", In Comrie, B. and Eid, M., editors, perspectives on Arabic Linguistics III: papers from the third Annual Symposium on Arabic Linguistics, pages 155 –172, John Benjamins, Amsterdam, 1991.
[19]. Ridley, M.J., "A code for bibliographic records transliterated from Greek", literary and Linguistic Computing, Vol. 7, pp. 27-29.
[20]. Knight, K. and J. Graehl, "Machine Transliteration", Proceedings of the 35th. Annual Meeting of the Association for Computational Linguistics, pp. 128-135, 1997.
[21]. Hussein K. AlKhafaji, Suhail A. Abdullah, Hanaa H. Merza, "A New Algorithm to Design and Implementation of Multilingual Spellchecker and Corrector", Journal of Al-Rafidain University College, ISSN 1681-6870, No. 32, 2013