Translating English Names to Arabic Using Phonotactic Rules *

Faisal Alshuwaier[a] and Ali Areshey[a]

Computer Research Institute, King Abdulaziz City for Science and Technology
P.O.Box 6086, Riyadh 11442, Kingdom of Saudi Arabia
{shuwaier, aareshey}@kacst.edu.sa

Abstract. With the increasing numbers of the arrivals to the Arabian countries; it is necessary to use Arabic language in writing names on official documents. Because of the difference in writing English names in Arabic language, many methods have been spread for translating English names which led to first, duplication in every single English name written in Arabic and this will lead to negative effects security and property rights. The second difference is that even if the transliteration of Arabic names is standardized, it is difficult for a layperson to implement it. This paper is to provide algorithms based on some rules that can be used in programming a system to transliterate English names automatically. The system uses only plan for Translate English Names to Arabic, and that can be processed and printed easily. Moreover, the Translated names can be read and recognized by ordinary people.

Keywords: Automatic Translation, Phonotactic Rules, Phonemes, Arabic, English, Diacritized Arabic Phone.

1 Introduction

With the rapid increase of the published information, it is difficult to rely on human translation to translate such information from one language to another and translation of appropriate names is generally realized as a significant issue in many multi-lingual text and speech processing applications. In our work, a set of hand-crafted transformations for locally editing the phonemic spelling of an English word to conform to rules of Arabic syllabification are used to seed a transformation-based learning algorithm. The algorithm examines some data and learns the proper sequence of application of the transformations to convert an English phoneme sequence to a Arabic syllable sequence. Our paper describes a data driven counterpart to this technique, in which a cascade model is used to go from English names to Arabic transliteration. The system uses only plan for Translate English Names to Arabic, and that can be processed and printed easily. Moreover, the Translated names can be read and recognized by ordinary people. This paper also discusses the problem background and provides solutions to Arabize almost all English Names. In section 2, we briefly highlight some related works. In section 3, we introduce description of translation system. In section 4, some specific phonotactic rules are shown. In section 5, the evaluation experiment for auto transliteration is presented. Finally, in Section 6, we draw some concluding remarks.

2 Related Works

Transliteration has been of interest to researchers in automatic lexicon creation and cross-language information retrieval involving radical differences in writing systems, such as English/Arabic, English/Korean, English/ Chinese and Japanese/English. (Knight and Graehl, 1998) have relied on the probabilistic mappings between English phones and Japanese phones generated. (Mansur

* The authors would like to acknowledge the reviewers for their valuable comments, which contributed to the clarity of the paper and in particular for their suggestions for the statements of rules.

et al., 1994) developed an algorithm at IBM for the automatic forward transliteration of Arabic personal names into the Roman alphabet. (Knight and Graehl, 1997) describe a back transliteration system for Japanese. (Lee et al. 1998) used a specific formula to generate a transliterated Korean word K for a given English word E. (Paola and Sanjeev, 2003) demonstrate the application of statistical machine translation techniques to translate the phonemic representation of an English name, obtained by using an automatic text-tospeech system, to a sequence of initials and finals, commonly used sub-word units of pronunciation for Chinese. (Stalls and Knight, 1998) present an Arabic-to- English back-transliteration system based on the source-channel framework.

3 Translation System Description

The procedure for constructing English-Arabic pairs is as follows: First, beginning with English name list which obtains each words English phonetic representations since these seemed to be most common English pronunciation symbol. Then the English pronunciation symbols are mapped to Arabic phonetic using a mapping table between the English phonetic characters. Then the Arabic phonetic sequences are mapped into diacritized Arabic name. Each diacritized symbol is mapped to the corresponding character. In Arabic language, the scripts are written right-to-left (RTL), so the system also manipulates the text direction. Finally, The Arabic name is represented without the Arabic diacritization.

3.1 English Name to English Phone

There are two general methods to mapping English names stored in the Database which contains over 133 thousand names to phonetic representations. The simplest one, if lexicon exists translating each name to its phonetic representations, then one can simply look up the corresponding phonetic stored in the Database which contains 84 symbols. However, this method will not work if a name is not existed. A more general method is to use a model of the English syllables to the phonetic sequences, and to choose the sequences that maximize the occurrence. In our model, we translated the names with known pronunciations from a pronunciation dictionary. We also used the Carnegie Mellon University (CMU) Speech Pronunciation Dictionary which applies 39 phonemes, each written as a sequence of Latin symbols (Carnegie Mellon University Project, 2007). We obtain the following English phonetic as an example for the name Obama:

$$\text{OBAMA} \longrightarrow \text{OW2-B-AA1-M-AH0}$$

Where each phone is separated by a hyphen "-". The CMU dictionary also provides the name representation since this seemed to be the most common pronunciation (Yan et al., 2003).

3.2 English Phone to Diacritized Arabic Phone

For mapping from an English phonetic existed in the CMU dictionary to an Arabic phonetic representation, we used phonotactic rules for mapping from English sounds to Arabic sounds. First, the system should map the English phonetic symbols to the Diacritized Arabic Phonetic symbols which a scan has been made for the universal phonetic symbols for (vowels), and linked it with the audio symbols for Vowels in Arabic so that we can convert the English names to the universal phonetic symbols, and then convert the universal phonetic symbols to the Arabic phonetic symbol. An obvious target inventory is the Arabic syllabary itself, shown in Table 1 as the short vowel symbols which are typically not written. Short vowels may be written with diacritics placed above or below the consonant that precedes them in the syllable, called *harakat* or *diacritics*. All Arabic vowels, long and short, follow a consonant; in Arabic, words like "Ali" or "alif".

In the fully vocalized Arabic text found in texts such as *Quran*, a long ā following a consonant other than a *hamzah* is written with a short a sign (*fatah*) on the consonant plus an *alif* after it; long is written as a sign for short i (*kasrah*) plus a ya ; and long ū as a sign for short u (*ammah*) plus a *waw*. Briefly, ^{a}a = ā, ^{i}y = $\bar{\text{i}}$ and ^{u}w =ū. Long ā following a *hamzah* may be represented by an *alif maddah* or by a free *hamzah* followed by an *alif* (Wikipedia).

Table 1: The values of Long and Short Vowels in Arabic

Symbols	Name	Trans.	Value
ـِ	kasrah	*i*	/i/
ـَ	fathah	*a*	/a/
ـُ	dammah	*u*	/u/
اَ	fathah alif	*ā*	/a:/
ىَ	fathah alif maqsura	*ā/aȳ*	/a/
وُ	dammah waw	*ū/ uw*	/u:/
يِ	kasrah ya	*ī/ iy*	/i:/

Table 2 provides the listing of the phoneme set and the corresponding phoneme symbols used in the system training . The table also shows illustrative examples of the vowel usage.

Table 2: The phoneme set

Arabic	Phoneme	Arabic	Phoneme	Arabic	Phoneme	Arabic	Phoneme
dammah/waw →ضمة/واو	/AO0/	fatha/alif → فتحة/ألف	/AA0/	Qaf → ق	/G/	Jeem → ج	/ZH/
dammah/waw →ضمة/واو	/UH0/	fatha/alif → فتحة/ألف	/AE0/	Qaf → ق	/R	Dal → د	/D/
dammah/waw →ضمة/واو	/UW0/	fatha/alif → فتحة/ألف	/AH0/	Kaf → ك	/K/	Daj → دج	/JH/
dammah/waw →ضمة/واو	/OY0/	fatha/alif → فتحة/ألف	/EH0/	Lam → ل	/L/	Thal → ذ	/DH/
dammah/waw →ضمة/واو	/OW0/	fatha/alif → فتحة/ألف	/EH2/	Meem → م	/M/	Reh → ر	/ER0/
dammah/waw →ضمة/واو	/UW1/	fatha/ya → فتحة/ي	/AY0/	Heh → هـ	/HH/	Reh → ر	/ER2/
wawya → وي	/OY1/	fatha/ya → فتحة/ي	/EY0/	Waw → و	/W/	Zain → ز	/Z/
Beh → ب	/B/	fatha/waw → فتحة/و	/AW0/	Noon → ن	/N/	Seen → س	/S/
Beh → ب	/P/	kasrah/ya → كسرة/ياء	/IH0/	Noon → ن	/Y/	Sheen → ش	/SH/
Ghain → غ	/NG/	Teh → ت	/T/	Feh → ف	/PH/	kasrah/ya → كسرة/ياء	/IY0/
Feh → ف	/F/, /V/	Teh Sh → تش	/CH/	dammah/waw →ضمة/واو	/UX/	kasrah/ya → كسرة/ياء	/IX/

The regular Arabic short vowels are /AE/. /IH/, and /UH/ corresponding to the Arabic diacritical marks Fatha, Kasra and Damma, respectively. The /AA/ is the pharyngealized allophone of /AE/. Similarly, the /IX/ and /UX/ are the pharyngealized allophones of /IH/ and /UH/ respectively. When /AE/ appears before an emphatic letter, its allophone /AH/ is used instead. The regular Arabic long vowel allophones are /AE:/ /IY/ and /UW/ respectively. The length of a long vowel is normally equal to two short vowels. The allophones /AY/ and /AW/ are actually two vowel sounds in which the articulators move from one post to another. These vowels are called Diphthongs. The allophone /AY/ appears when a Fatha comes before an undiacritized Yeh. Similarly, /AW/ appears when a Fatha comes before an undiacritized Waw. The Arabic voiced stops phonemes /B/ and /D/ are similar to their English counter parts. /DD/ corresponds to the sound of the Arabic Dhad letter. The Arabic voiceless stops /T/ and /K/ are basically similar to their English counter parts. The sound of the Arabic emphatic letter Qaf is represented by the phone /Q/. The Hamza plosive sound is represented by the phone /E/, and the sound of Jeem (in many dialects) is represented by /G/. The voiceless fricatives are produced with no vibration of the voice cords (Mohamed, et al., 2007). The set of symbols is called International Phonetic Alphabet (IPA) (International Phonetic Association, 2005) which its symbol is based on the Roman letters as shown in Figure 1.

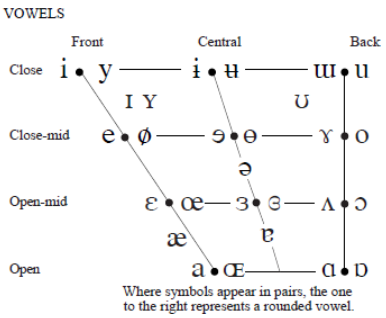


Figure 1: International Phonetic Alphabet

In addition, IPA does not have symbols for the emphatic sounds:

ص ـ ض ـ ط ـ ظ

So, to make it possible for researcher and Arab linguists and who work on language sounds, a new alphabet has been created. Arabic International Phonetic Alphabet (AIPA) consists of symbols that are based on Arabic orthographic system. (Mansour Alghamdi et al., 2004; Wikipedia; Alghamdi, 2003). Figure 1 presents the phonetic symbols as they are the IPA which consists of 28 sounds, however, in Arabic has 3 sounds symbols as:

إ، أ ، ا

For the phonetic sequence OW2-B-AA1-M-AH0, this step creates the diacritized Arabic sound sequence as follows:

OW2-B-AA1-M-AH0 ⟶

اَ ـ م ـ أَ ـ ب ـ و ـ اُ

3.3 Diacritized Arabic Phone to UnDiacritized Arabic Phone

For mapping from a diacritized Arabic phone representation to undiacritized Arabic phone representation, we use traditional mappings between the diacritized symbols, corresponding characters as shown in Table 1 and the phonetics rules that will be explained in the next section. Because short vowels (diacritics) are commonly not presented in Arabic orthography, this creates a problem in transliterating unknown proper names (Out Of Vocabulary) since these missing diacritics should be deduced before transliteration to obtain the appropriate pronunciation. For the phonetic sequence:

اَ ـ م ـ أَ ـ ب ـ و ـ اُ

This step creates the undiacritized Arabic sound sequence as follows:

أ و ب ا م ا ⟶ أَ م أَ ب و اُ

The steps in English-to-Arabic transliteration of names are depicted in Figure 2.

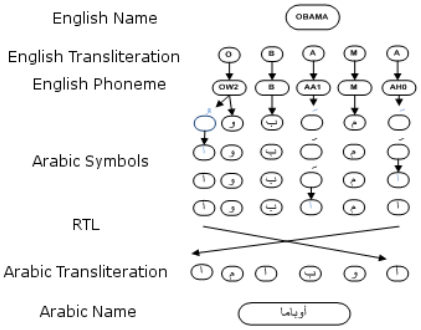


Figure 2: The steps in English-to-Arabic transliteration of names

4 Phonotactic Rules

Using the selected phoneme set, we developed a set of rules that are used to automatically generate the phonetic pronunciations for Arabic words. Each rule tries to match certain conditions on the context of the letter and provide a replacement from the phoneme list. Replacements can be one or more phonemes. Some letters dont have an effect on pronunciation or, depending on context, they might not be pronounced; in this case, the replacement will be empty (Mohamed et al., 2007). Each rule follows this format:

/ # (pre_rule) ⟶ / # (post_rule)

or

(pre_rule) # / ⟶ (post_rule) # /

To facilitate understanding of the Phonotactic Rules, the Table 3 explains the symbols used in. The left hand side of the rule is pre-replacement and each letter in the pre-replacement is referenced by its name as defined in the Unicode standard. Multiple classes are defined to simplify the rules syntax. The right hand side of the rule defines the post-replacement, which can either be a phoneme or a sequence of phonemes from the phoneme list, or the letter might not have a matching phoneme and will be omitted from pronunciation.

Table 3: Symbols used in the Phonotactic Rules

Symbol	Significance	Symbol	Significance
/ #	Beginning of the word	( )	Character between brackets optional
# /	End of the word	[ ]	It corresponds to another character
C & V	Consonant & Short Vowels	⟶	Become
_	Character of the applicable rules	Cn	A series of consonants

4.1 First Rule

Short vowels can be normal ones or either emphatic or pharyngeal, depending on the surrounding letters. We developed rules that take care of all these situations. If the name starts with Arabic diacritization it will be transferred to corresponding Characters. For instance, Fatha will be transferred to Fathah Alif as described in the condition 1, Dammah will be transferred to Dammah Alif as shown in the condition 2 and Kasrah will be transferred to Kasrah Alif as described in the condition 3.

/# [◌َ] → /# [ٰا] (1) /# [◌ُ] → /# [ا] (2) /# [◌ِ] → /# [ا] (3)

4.2 Second Rule

The vowels (A, E, I, O, U and sometimes Y) are sometimes treated as semi-vowels and other times it is treated as a long vowel, depending on its context. The condition 4 describes the second rule. If the name ended with one of these vowels will be transmitted to the following letters depending on the pronunciation of the letter in Arabic : ”Alif” or ”Waw” or ”Ya”.

_v #/ → _ [ا / و / ي] #/ (4)

4.3 Third Rule

In the case of a vowel at the beginning of the word optional (v) followed by consonant and then a vowel v as shown in the condition 5, the second vowel to be switched to the following letters depending on the pronunciation of the letter in Arabic : ”Alif” or ”Waw” or ”Ya”.

/# (v) c v → /# (v) c [ا / و / ي] (5)

4.4 Fourth Rule

In the case of a vowel at the beginning of the word optional (v) followed by consonant and then a vowel v as shown in the condition 6, the second vowel to be switched to the following letters depending on the pronunciation of the letter:

(أ ، و ، ي)

$$/\#\ (v)\ c_n\ \underline{v} \rightarrow /\#\ (v)\ c_n\ \underline{[ا\ و\ ي]} \qquad (6)$$

4.5 Fifth Rule

In the case of a vowel at the beginning of the word optional (v) followed by consonant and then a vowel v as shown in the condition 7, the second vowel to be switched to the following letters depending on the pronunciation of the letter in Arabic : ”Alif” or ”Waw” or ”Ya”.

$$/\#\ (v)\ c\ \underline{NG}\ \#/ \rightarrow /\#\ (v)\ c\ \underline{نغ}\#/ \qquad (7)$$

4.6 Sixth Rule

In the case of a vowel at the beginning of the word optional (v) and then followed by the letter N and G and then a fixed character as shown in the condition 8, the letters (NG) together are switched to the following:

(ن)

$$/\#\ (v)\ \underline{NG}\ c \rightarrow /\#\ (v)\ \underline{نـ}\ c \qquad (8)$$

4.7 Seventh Rule

In the case of a vowel at the beginning of the word optional (v) and then fixed character C1 followed by fatah or dammah or kasrah and then followed by same fixed character C1 as shown in the condition 9, then fatah or dammah or kasrah will be converted to the following character:

إِ ، أُ ، أَ

$$/\#\ (v)\ c1\ \underline{[\ \ ً\ \ ٌ\ \ ٍ\ ]}\ c1 \rightarrow /\#\ (v)\ c1\ \underline{[ا\ و\ ي]}\ c1 \qquad (9)$$

4.8 Eighth Rule

In the case of a vowel at the beginning of the word optional (v) and a fixed character C1 followed by fatah or dammah or kasrahand then followed by same fixed character C1 as shown in the condition 10, only one character from C1 to be written in addition to:

·اّ

$$/\#\ (v)\ c1\underline{[\ \ ً\ \ ٌ\ \ ٍ\ ]}\ c1 \rightarrow /\#\ (v)\ c1[\ \underline{ّ}\ ] \qquad (10)$$

5 Evaluation Experiment

In order to evaluate the correctness of our system in term of agreement and ordering handling, we have developed an evaluation methodology based on a comparison between the system outputs with the original translation of the input text. The following steps describe the evaluation methodology:

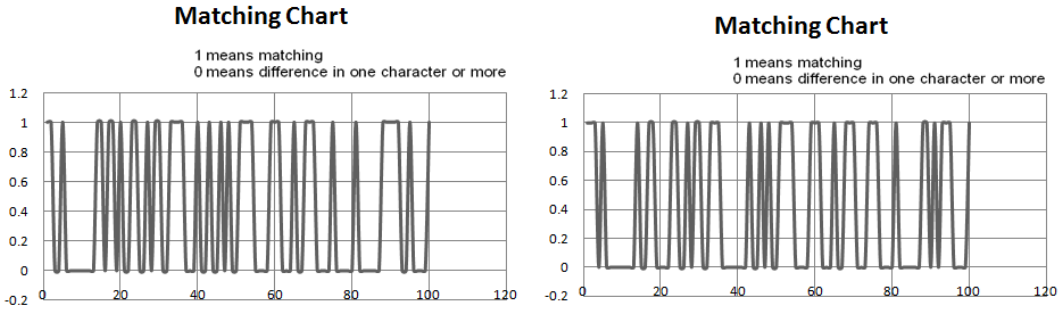
(1) Run the system. (2) Compare the original translation with the system output. (3) Classify the problems that arise from the mismatches between the two translations. (4) Assign a suitable score for each problem. A range of score between 0 and 10 determines the correctness of the translation. While 0 indicates absolutely incorrect translation and 10 indicates absolutely correct (matched) translation. (5) When a situation belongs to multiple problems compute its score average. (6) Determine the correctness of the test case by computing the percentage (%) of the total scores (T.S.).

5.1 Extracting Named Entity Transliteration Experiment

The purpose of this experiment is to investigate whether the following machine translation systems (MTS), namely, Google, Bing and our system, are sufficiently robust for handling the word agreement and ordering in the translation the names from English to Arabic. The evaluation methodology is applied on 100 independent test examples. The experiment gives the following results between our system and Google & Bing as shown in Table 4. Figure 3 presents the matching chart bet. Google (a) and Bing (b) with our system.

Table 4: Experiment Results bet. System and Google & Bing

MTS	Matches	Mismatches	T.S. of Matches	T.S. of Mismatches	T.S.	%
Google	41	59	410	405	815	81.5%
Bing	40	60	400	421	821	82.1%



(a) Google with Our System

(b) Bing with Our System

Figure 3: Matching Charts

The percentage of the total score for each system has been found by dividing the total score by 1000; as we have 100 test examples and each is evaluated out of 10. The mismatches test examples have problems that arise from the mishandling the word agreement and ordering in the target language. The following steps classifies these problems and assigns suitable scores for them; we have classified the problems to either agreement or ordering problems as following: (1) Difference in one symbol is evaluated out of 9. (2) Difference in two symbols is evaluated out of 8. (3) Difference in three symb. is evaluated out of 7. (4) Difference in four symb. is evaluated out of 6.

6 Conclusion

We proposed an English-Arabic transliteration model using phonetic rules and pronunciation. We provided algorithms based on phonetic rules that can be used in programming a system to transliterate English names automatically. Since Arabic transcription is playing an increasingly important role in a variety of practical applications, it is necessary to pursue efforts to develop more language-specific transcription systems based on linguistic knowledge.

References

Algamdi, Mansour. 2003. *KACST Arabic Phonetics Database.* The Fifteenth International Congress of Phonetics Science, Barcelona, 3109-3112.

Carnegie Mellon University Project. *Cmusphinx, Free Pronouncing Dictionary of English*. `https://cmusphinx.svn.sourceforge.net/`

Helen M. Meng, Wai-Kit Lo, Berlin Chen, and Karen Tang. 2001. *Generating Phonetic Cognates to Handle Named Entities in English-Chinese Cross-Language Spoken Document Retrieval.* Proceedings of ASRU.

International Phonetic Association. *THE INTERNATIONAL PHONETIC ALPHABET*. `http://www.langsci.ucl.ac.uk/ipa/IPA_chart_(C)2005.pdf/`

Kevin Knight and Jonathan Graehl. 1997. *Machine transliteration.* In Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics, pages 128-135.

Khaled Shaalan. 2004. *Machine Translation of English Noun Phrases into Arabic*. International Journal of Computer Processing of Oriental Languages Vol. 17, No. 2.

Knight, K. and Graehl, J. 1998. *Machine Transliteration.* Computational Linguistics: 24(4).

Lee, J. S. and K. S. Choi. 1998. *English to Korean Statistical transliteration for information retrieval.* Computer Processing of Oriental Languages,12(1):17-37.

Mansour M. Alghamdi. *Arabic International Phonetic Alphabet.* `http://www.mghamdi.com/AIPA%20.pdf/`

Mansour Alghamdi, Husni Almuhtasib and Mustafa Elshafei. 2004. *"Arabic Phonological Rules"*. King Saud University Journal: Computer Sciences and Information. Vol. 16, pp. 1-25.

Mansur Arbabi, Scott M. Fischthal, Vincent C. Cheng, and Elizabeth Bart. 1994. *Algorithms for Arabic name transliteration.* IBM Journal of Research and Development, 38(2):183-193.

Mohamed Ali, Moustafa Elshafei, Husni Al-Muhtaseb, and Mansour Al-Ghamdi. 2007. *Automatic Segmentation of Arabic Speech.* Workshop on Information Technology and Islamic Sciences, Imam Mohammad Ben Saud University, Riyadh, March.

Paola Virga and Sanjeev Khudanpur. 2003. *Transliteration of proper names in cross-lingual information retrieval*. MultiNER '03 Proceedings of the ACL 2003 workshop on Multilingual and mixed-language named entity recognition - Volume 15.

Rached Zantout and Ahmed Guessoum. 2000. *Arabic Machine Translation: A Strategic Choice for the Arab World*, Vol. 12, Comp. and Info Sci., pp. 117-144. King Saud University, KSA.

Soudi A., Bosch A. and Neumann G. 2007. *Arabic Computational Morphology: Knowledge-based and Empirical Methods, Text and Language Technology*. Vol 38. Springer, New York.

Stalls, Bonnie G. and Kevin Knight. 1998. *"Translating Names and Technical Terms in Arabic Text"*. In Proceedings of the COLING/ACL on Comput. Approaches to Semitic Lang. 19.

Sung Young Jung, SungLim Hong, and Eunok Paek. 2000. *English to Korean Transliteration Model of Extended MarkovWindow.* Proceedings of COLING.

Yan Qu, Gregory Grefenstette and David A. Evans. 2003. *Automatic Transliteration for Japanese-to-English Text Retrieval.* SIGIR03, July 28August 1, 2003, Toronto, Canada.