arab journal for the humanities

189

Issue 89 / 23

A Computational Approach to Arabic Orthographic Relaxation

Salah Alnajem

Lecturer, Department of Arabic,
College of Arts, Kuwait University

Abstract

This paper introduces a finite-state computational approach to Arabic orthographic *relaxation*. Relaxation is a technique which allows computational systems to handle human errors caused by orthographic variations existing in the Arabic orthographic system. The impact of such variations is considered in addition to exploring positive and negative aspects of the relaxation technique, which will be used to handle such variations.

1. Introduction

Arabic orthography exhibits a number of orthographic variations in which abstract lexical words differ from surface orthographic forms. The main variations in Arabic orthography are related to the writing of the glottal stop and the weak letters. These variations normally cause writing errors and confusion even among native speakers of Arabic. They have also an impact on computational systems dealing with Arabic texts like database management systems.

In this paper we discuss this phenomenon through considering the variations in the orthography of the verb-initial glottal stop, and the variations in the orthography of verb-final weak letters. Next, the impact of this phenomenon is considered. Then, the paper introduces a finite-state computatioal approach to Arabic orthographic relaxation. Relaxation is a technique which allows computational systems to handle human errors caused by such orthographic variations. This approach is part of a larger computational approach to the derivation and inflection of Arabic verbs (Alnajem, 2002a, 2002b, 2002c). Positive and negative aspects of the relaxation techniques are also explored.

2. The Phenomenon

The word-initial glottal stop, known as Hamza, has special conditions in the orthographical system of Arabic. It has two representations illustrated in (1):
(1)
a- أكل الولد التفاحة
b- استعمل الولد المفتاح
The word-initial glottal stop can be written as أ [1] or as ا . The first orthographic representation of the glottal stop is known as the Disjunctive Hamza همزة قطع. The second is known as the Conjunctive Hamza همزة وصل. The Conjunctive Hamza is a glottal stop which is pronounced only if preceded by a pause. It is not pronounced if it is preceded by a word or particle. The Cut Hamza is a glottal stop which is always pronounced. Thus, we have one grapheme (Hamza) but with two orthographic representations when this grapheme is used in a word-initial position. Alnajem (2002d) discussed the variation in the orthography of verb-initial glottal stop, and introduced a linguistically-motivated account for these variations. Alnajem (2002d) introduced also a computational approach to handle such variations.

Weak letters have different orthographic representations in the orthographical system of Arabic according to their position in the word. In this paper we consider the relaxation of the variations in the orthography of verb-final weak letters. The verb which contains among its radical letters a weak letter or more than one weak letter is called a Weak Verb. This weak letter comes either at the beginning, middle, or end of the verb. An example of verbs whose first radical letter is a weak letter is وعد . The verb قال is an example of a verb which has a medial weak letter. An example of verbs with a final weak letter is دَنَوَ . Verbs with an initial or medial weak letter undergo phonological variations but not orthographic. Such variations have been considered in Alnajem (2002b). The orthographic variations are applied in the case of the verbs which end with a weak letter.

The last radical letter of the verb which ends with a weak letter can either be /w/ or /u/ (glide). This means that, at the lexical abstract level, the verb which ends with a final weak letter ends with a glide. At the orthographic surface level, this glide may be changed to Extended Alef ألف ممدودة which is written ا as in:

(2) دَعَوَ → دَعا

or as a Maqssora Alef ألف مقصورة written ى as in:

(3) رَمَيَ → رَمى

In the case of verbs derived from pattern I (CVCVC), traditional grammarians say that the shape of the Alef (Extended or Maqsoura) is governed by the original weak letter from which the Alef has been transferred. The Alef is written as an Extended Alef if the original weak letter was ‘w’ as in (2). The Alef is written as a Maqsoura Alef if the original weak letter was ‘y’ as in (3) (Alkhatib, 1994).[2]

In the case of verbs derived from the other patterns, the Alef is written as Maqsoura as in:

(4) أعطيَ → أعطى (Pattern IV)

except in the verbs whose penultimate letter is ‘y’. In this case, the Alef is written as Extended as in:

(5) استحيَيَ → استحيا (Pattern X)

These variations have been studied by traditional grammarians who suggest a number of rules which govern them. Many of these rules are not well understood even by the native speakers of Arabic since they are complicated to some extent. Even specialists do not fully understand these rules. Besides, many of these rules are descriptive rules and not linguistically-motivated to the standards of modern linguistics.

This has caused the widely encountered errors which appear in the writing of native speakers of Arabic. The impact of the orthographic errors affects also the Information Technology field. In database systems, a record in a database may contain a misspelled value like the proper noun احمد. The correct orthographic form of this noun is أحمد . When a query is sent to this database with the correct formأحمد, the database will not return, as a result, the record containing the misspelled name أحمد. This will cause loss of data and confusion, especially in mission-critical database systems like those related to government. Another instance relating to the IT field is the search engines. The problems faced in database systems are frequently faced in web-search engines, which are programs connected to databases. These problems are still facing the IT industry, and have not been solved yet even by the major database management system vendors.

3. Orthographic Relaxation

Orthographic relaxation is a technique which is used to allow a software to accept the misspelled input of the user as a correct input without forcing that user to correct the orthographic errors. A user of an e-mail client software supplied with an Arabic orthographic relaxation technique can write an e-mail message with misspelled words like:

(6)
a-* اكل
b-* رما
c-* دعى

without marking these words as misspelled and without forcing the user to correct these words to:

(7)
a- أكل
b- رمى
c- دعا

A database system supplied with an Arabic orthographic relaxation technique can accept from the user the misspelled value اكل or the correct value أكل as a value for a textual field during data entry stage. After entering the data in the database, that system can accept from the user a query like اكل, and it automatically searches for the correct form أكل while searching also for the misspelled entered form اكل.

Orthographic relaxation can also be used to apply Automatic Correction for the frequently encountered orthographic errors without forcing the user to manually correct them. These automatic corrections are applied in Real Time (as soon as they are encountered by the system). For example, an Arabic word processor supplied with an Arabic orthographic relaxation technique can automatically correct the errors of writing verb-initial glottal stops and verb-final glides as soon as these errors are encountered. A word like اكل can be automatically corrected to أكل in real-time, and a word like دعى can be automatically corrected to دعا.

Orthographic relaxation can also be used to deal with vocalization in Arabic. In Arabic, a word is written with full, partial, or with no vocalization. In this context, a user may enter his data to a data base with full vocalization. In the query stage, a query can be entered with full, partial, or with no vocalization. The relaxation technique can search for all the possible vocalizations of a query entered with a partial or full vocalization. For example, a query like كتب with no vocalization can return كَتَبَ ، كتيبَ, and other verbs.

3.1 Positive and Negative Aspects

Orthographic relaxation has positive and negative aspects from the linguistic and the computational point of view. One of the main positive aspects of the relaxation technique is that it eliminates the problem of data loss and confusion resulting from entering misspelled values to database systems. The problem mentioned at the end of section two will not occur with a database system supplied with the orthographic relaxation technique.

Orthographic relaxation technique also allows us to supply database systems with the ability to search for values similar to the entered value in terms of orthography. This makes the database more useful, and facilitates the query process. In this context, the user can enter a query with the value رعى, and the database system can return results with values orthographically similar to the entered query value.

Orthographic relaxation technique takes into account the increase of writing errors in environments which require fast response by writing. Examples of such environments are electronic chat rooms, web fora, and e-mail client programs. The users of such systems normally fall into typing errors which are more than the errors of a user of other applications. Automatic real time correction saves the user in such environments from spending time in correcting the orthographic errors. It saves the reader also from the comprehension problems caused by such errors.

Besides, orthographic relaxation allows the software to deal with the different levels of vocalization (full, partial, and non-vocalized forms). Handling the different levels of vocalization represents a challenge to Natural Language Professing systems. A word processor supplied with Arabic relaxation can check the spelling of fully, partially, or non-vocalized words. When it faces a misspelled word (like كُتَبا) it takes one of the following two actions according to the settings adjusted by the user during software installation. It can accept the misspelled word after removing the wrong vocalization to give the bare correct form (كتبا). Otherwise, it can suggest possible vocalizations. For example, when such a word processor checks the word كُتَبا it marks it as an unacceptable word, and it suggests the possible vocalizations for this word which include:

(8)
a- كَتَبا
b- كُتِبا

Checking the spelling of vocalized Arabic words is a difficult task. This is the reason why the Arabic spell checker of the most popular word processor Microsoft Word XP does not handle Arabic vocalization during its spell checking. When you write the following misspelled words, it accepts them as correct words since it ignores vocalization:

(9)
a-* كَتُبا
b-* كتَبا
c-* كتبا

On the other hand, orthographic relaxation has some negative aspects. Linguistically, relaxation can decrease a user's language awareness. The user may not focus on writing words using their correct spelling because he/she knows that the software will know what he/she means or because he/she knows that the system will automatically correct his/her typing errors.

Automatic correction can affect language competence among language learners who frequently use word processors supplied with automatic correction to write their documents. Writing orthographic errors without getting feedback that what they are writing is wrong can affect their knowledge of the orthographical system of a language.

In addition, relaxation can cause confusion when a relaxation system returns to the user as an output a number of alternatives for a single misspelled word. For example, when a user enters the word قُتلا the system can return the following list of alternative spellings in addition to others:

(10)
a- قَتَلا
b- قُتلاً
c- قَتَلاً

4. A Computational Approach to Arabic Orthogrpahic Relaxation

This section introduces a computational approach to Arabic orthographic relaxation. This approach is part of a larger computational approach to Arabic verb morphology. This approach is used to produce a morphological analysis and generation system for Arabic verb morphology. The analysis and generation system will be expanded later to cover the Arabic noun morphology. The approach will be used to design a robust Arabic spell checker and will be used as a component for larger natural language processing systems. Aspects of this computational approach to Arabic verb morphology have been covered in Alnajem (2002a, 200b). In this paper we consider how this approach can be used to handle the relaxation of Arabic verb orthography. The relaxation of Arabic noun orthography is handled in a similar way.

4.1 Finite-State Morphology

The proposed computational approach is a finite-state approach, which is implemented using Xerox Finite-State Tools. It is an approach which uses the computational power of Finite-State Automata and the power of the mathematical operations which are applied to these automata. For a formal description of finite-state automata, see Roche and Schabes (1997). Finite-state automata are appreciated in the literature for their mathematical power. Accordingly, mathematical terms and operations are normally used to describe and manipulate these automata. The main mathematical objects used to manipulate finite-state automata are set operations like union, subtraction, and composition. Xerox Finite-State Tools and Zerox Finite-State Calculus use such operations to describe and manipulate finite-state automata, and we use them to deal with Arabic morphology.

Finite-state automata are widely used in Arabic computational morphology. It represents the heart of Arabic morphology modeling. The majority of systems proposed in the literature to handle Arabic morphology computationally are finite-state systems. These include systems proposed by Kay (1987), Beesley (1991, 1996, 1998), Kiraz (1994, 2001), Kornai (1991), Bird and Ellison (1992, 1994), and Narayanan and Hashem (1993).

4.2 Derivation and Inflection

Verb derivation is achieved in our approach in a way which focuses on capturing dependencies between patterns. This is done through the use of finite-state transducers and mathematical operations applied to these transducers. The operations are union, subtraction, and composition. The derivation process yields inflectional stems derived from roots using patterns.

Table 1 shows some of these inflectional stems derived using a number of patterns. These inflectional stems are used to form perfective (active and passive), imperfective (active and passive), and imperative verbs through the affixation of inflectional suffixes or prefixes and suffixes. The stems are given in this table without inflectional prefixes and suffixes. For the triliteral verbs, the root *ktb* (writing) will be used as an example root; while the root *dhrj* (rolling) will be used for the quadrilateral verbs. Roman numbers are used for reference, with the prefix Q denoting a quadrilateral verbal pattern. Not all the patterns occur with every root. The root *ktb*, for example, does not occur with some patterns. For instance, **nkatab* (Pattern VII) is unacceptable in Standard Arabic while *katab* (Pattern I) is acceptable. It should also be noted that some stems undergo certain additional phonological processes. For example, the stem *staktab* (Pattern X) has the surface form /?i/ *staktab* with the epenthetic syllable /?i/ prefixed to it. This epenthetic syllable /?i/ will be mentioned in the table in parenthesis. In addition, in this table we will use CV arrays to represent the perfective, imperfective, and imperative stems (patterns) of each pattern.

Table 1: Verbal patterns

No	Pattern Name	Active Perfective	Passive Perfective	Active Imperfective and Imperative	Passive Imperfective
I	fa9al	$C_{1a}C_{2a}C_3$ katab	$C_{iu}C_{2i}C_3$ kutib	$C_1C_{2u}C_3$ ktub	$C_1C_{2a}C_3$ ktab
II	fa99al	$C_{1a}C_2C_{2a}C_3$ kattab	$C_{1u}C_2C_{2i}C_3$ kuttib	$C_{1a}C_2C_{2i}C_3$ kattib	$C_{1a}C_2C_{2a}C_3$ kattab
III	faa9al	$C_{1aa}C_{2a}C_3$ kaatab	$C_{1uu}C_{2i}C_3$ kuutib	$C_{1aa}C_{2i}C_3$ kaatib	$C_{1aa}C_{2a}C_3$ kaatab
IV	?af9al	$?_aC_1C_{2a}C_3$?aktab	$?_uC_1C_{2i}C_3$?uktib	$?_aC_1C_{2i}C_3$?aktib	$?_aC_1C_{2a}C_3$?aktab
QI	falal	$C_{1a}C_2C_{3a}C_4$ dahraj	$C_{1u}C_2C_{3i}C_4$ duhrij	$C_{1a}C_2C_{3i}C_4$ dahrij	$C_{1a}C_2C_{3a}C_4$ dahraj
QII	tafa9lal	${}_{ta}C_{1a}C_2C_{3a}C_4$ tadahraj	${}_{tu}C_{1u}C_2C_{3i}C_4$ tuduhrij	${}_{ta}C_{1a}C_2C_{3a}C_4$ tadahraj	${}_{ta}C_{1a}C_2C_{3a}C_4$ tadahraj

After constructing the inflectional stems, the inflectional component of the approach is used to construct fully inflected verbs from the inflectional stems through the affixation of suffixes or prefixes and suffixes (circumfixes) which indicate person, number, and gender. In this affixation process, the long-distance dependency between the circumfixes for the imperfective verbs is handled through composition. The inflectional component of the approach handles the inflection of verbs in Arabic in a way which captures generalizations, sub-generalizations, and syncretisms which govern verb inflection. This component yields fully inflected verbs like those in the following perfective active inflectional paradigm:

Table 2: The perfective active paradigm

	Mas	**Fem**
Singular		
1st	katab-tu	katab-tu
2nd	katab-ta	katab-ti
3rd	katab-a	katab-at
Dual		
1st	katab-naa	katab-naa
2nd	katab-tumaa	katab-tumaa
3rd	katab-aa	katab-ataa
Plural		
1st	katab-naa	katab-naa
2nd	katab-tum	katab-tunna
3rd	katab-uu	katab-na

Some of these verbs are still abstract and do not represent a surface form. Further changes in verb structure like radical letter rejection or transformation are also applied in the inflection process when we deal with Weak Roots. Weak Roots are roots which contain among their radical letters a weak letter or more than one weak letter. Furthermore, some verbs undergo epenthetic prefixation as previously mentioned. Such changes have been considered in Alnajem (2002b).

In this approach, these phonological/morpho-phonological variations are applied through special rules known as Replace Rules. These rules are automatically converted to finite-state transducers by Xerox Finite-State Tools. These rules are formalized in a linguistically-motivated manner and they are similar to what is known as context-sensitive phonological re-writing rules. They are written as follows:

A -> B || L _ R

This rule states that any upper-sided string containing a string from A is related to a string or strings on the lower-side containing a string from B. This mapping is applied when the left context ends with L and the right context begins with R.

A replace rule is written for each phonological/morpho-phonological variation. A replace rule can be optional or obligatory. The rules are applied in parallel or in a sequential ordering where the output of a rule serves as an input for the next rule. The replace rules are applied to words through a mathematical operation widely applied in the field of finite-state computing which is composition. The replace rules are composed with the lower side of the lexicon transducer which holds the inflected verbs as its lower side language, and the morphosyntactic feature strings which correspond to those verbs as its upper-side language. This composition produces a new lexicon transducer whose upper-side language consists of morphosyntactic feature strings, and its lower-side language consists of fully inflected verbs corresponding to the morphosyntactic feature strings after applying the required variation rules.

4.3 The Relaxation of Arabic Verbal Orthography

In this paper we will consider how the orthographic relaxation is applied to deal with three types of orthographic phenomena: the orthography of the verb-initial glottal stop, the orthography of verb-final glides, and the orthography of vocalization. The relaxation of other orthographic phenomena can be handled in a similar way. Alnajem (2002c) gives more details about the orthography of verb-initial glottal stops and the orthography of verb-final glides.

4.3.1 The Relaxation of the verb-initial glottal stop

In Arabic orthography, the word-initial glottal stop (Hamza) has a special nature. It has two shapes (variants) illustrated in (11):

(11)

a- أكل الولد التفاحة

b- استعمل الولد المفتاح

From (11) the reader can understand that the word-initial glottal stop is written as أ or ا. The first word-initial glottal stop is known as Disjunctive Hamza همزة قطع. The second initial glottal stop is known as the Continuous

Hamza همزة وصل. The first type of initial glottal stop is defined by the traditional grammarians as the initial glottal stop which is not deleted in continuous speech (Alduqur, 1986). The continuous initial glottal stop is a glottal stop which is written but not pronounced in continuous speech (Assi and Ya'aqoub, 1987).

What governs the variation in writing the word-initial glottal stop is a phonological phenomenon related to the epenthetic short syllable /?V/, which is prefixed to the verbs and nouns beginning with a CC consonant cluster. The Continuous Hamza (ا) is used in the verbs and nouns which begin with a CC consonant cluster. When this CC consonant cluster is prefixed by the epenthetic /?V/ syllable, the onset (?) is written as the Continuous Hamza. If the verb or noun is preceded by a pause, the onset (?) is pronounced as an ordinary glottal stop. If the verb or noun is not preceded by a pause (in continuous speech) the epenthetic /?V/ syllable is not pronounced and the first consonant in the CC consonant cluster becomes the coda of a closed syllable beginning with the last short syllable in the preceding word/article (as shown in Fig. 1) If the preceding word/article ends with a closed CVC syllable, a short vowel is inserted after the coda of this closed syllable, which makes the coda of this closed CVC syllable an onset of a short syllable. That short syllable is integrated with the first consonant of the CC consonant cluster to form a closed syllable (as shown in Fig. 2). A phonological re-writing rule which describes the deletion of the epenthetic /?V/ syllable can be formalized as follows:

$$?V \rightarrow 0 \,/\, C\ V \underline{\quad\quad} [_{Stem}CC$$

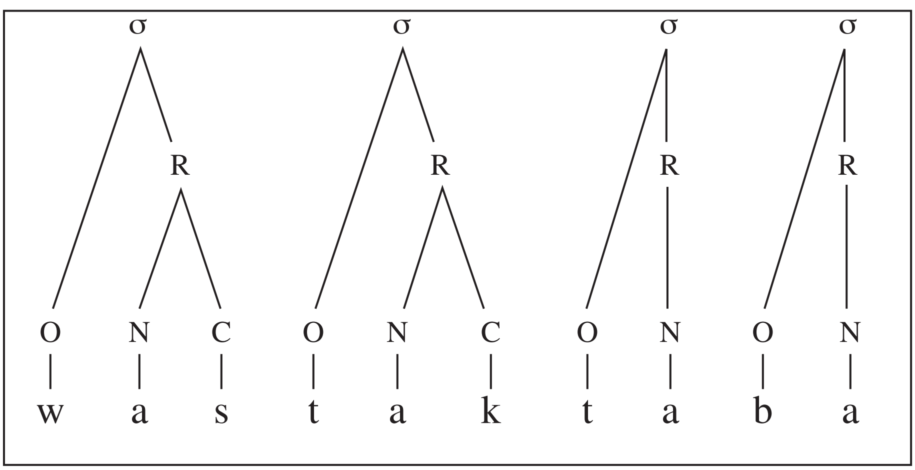

Fig. 1: The syllabic structure of *wa (?i) staktab-a*

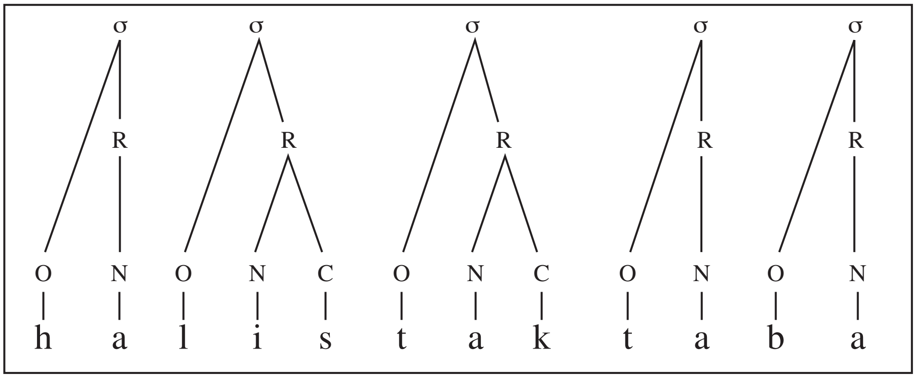

Fig. 2: The syllabic structure of *hal (?i) staktab-a*

Whether the CC consonant cluster is preceded by a pause or not, the verb-initial glottal stop is represented orthographically as Continuous Hamza.

On the other hand, the Disjunctive Hamza (أ) and its variant (إ) occur in words which do not begin with a CC consonant cluster. In other words, the Disjunctive Hamza occurs in words which do not undergo the epenthetic /?V/ prefixation.

Examples of words beginning with a Disjunctive Hamza are the verb أَكَلَ (to eat, Perf Act Third Sing Mas) and the noun أقلامٌ (pens, P1 Mas Indefinite) which have the following syllabic structures:

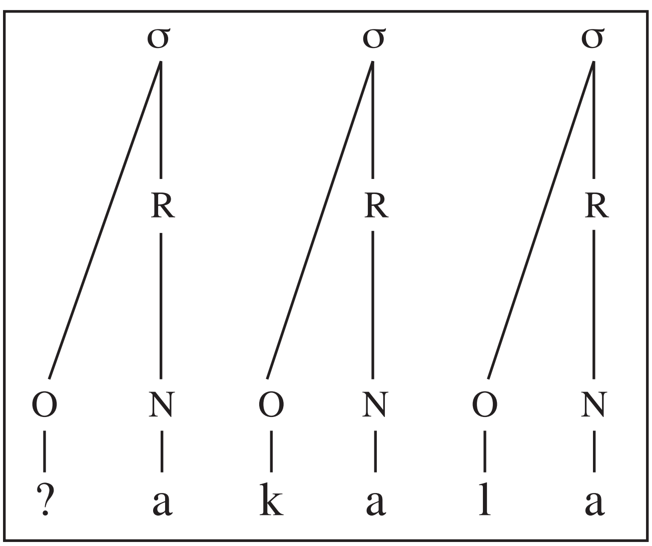

Fig. 3: The syllabic structure of *?akal-a*

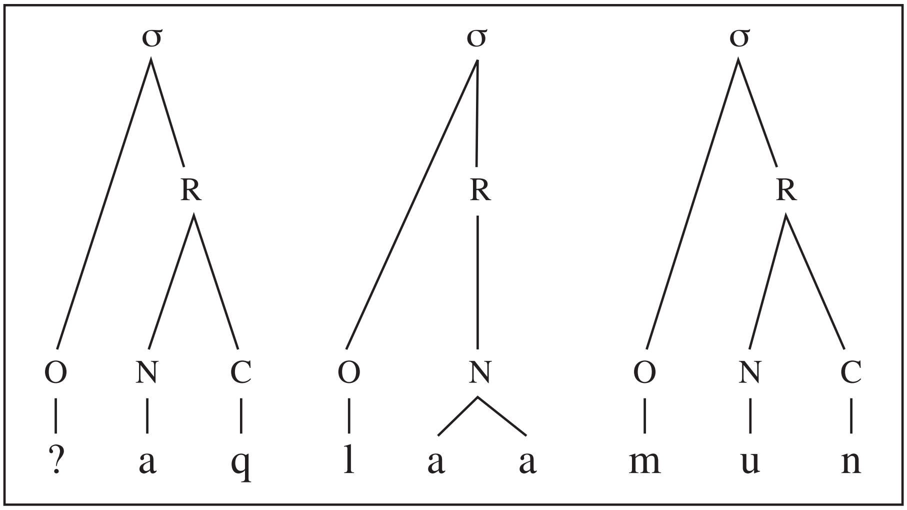

Fig. 4: The syllabic structure of *?aqlaam-un*

Even in the continuous speech, the word-initial glottal stop in the verbs and nouns above is not deleted. Stem medial glottal stop is always written as Disjunctive Hamza as in سأل.

The replace rule which allows for the relaxation of the Continuous Hamza orthography is:

[%! (–>) %@ || .#. _ [V–a] C C] ;

This rule says that the Continuous Hamza (!) can optionally be written as a Disjunctive Hamza (@) or it can be kept without change. The (->) is a special symbol which means the optional application of a rule. The symbol '%' is a special symbol written before using the symbols reserved by Xerox Finite-State Tools which include (!, ad @). This rule will allow the system to accept verbs like * أكتبا (with the application of the optional rule) and اكتبا (without applying the optional rule). Notice that the nucleus of the initial syllable should be either the vowel 'i' or 'u'. Excluding the vowel 'a' eliminates the case of pattern IV. In this case the /?V/ sequence is not epenthetic and the glottal stop is written as a Disjunctive Hamza.

The replace rule which allows the relaxation of the orthography of the Disjunctive Hamza is:

[%@ (–>) %! || .#. _ V \ [C C]] ;

This rule says that the Disjunctive Hamza (@) can optionally be written as a Continuous Hamza (!) or it can be kept without change. This will allow the system to accept verbs like * اكَلَ (with the application of the optional rule) and أكَلَ (without applying the optional rule). This rule is applied to verbs beginning with a /?V/ sequence followed by any sequence except a Cc consonant cluster. The symbol '/' is used to indicate exclusion. This context excludes the case which triggers the prefixation of the epenthetic /?V/ syllable. This case is the only case in which the Conjunctive Hamza is used. In the context of this rule, the /?V/ sequence is not epenthetic and the glottal stop is written as a Disjunctive Hamza.

4.3.2 The Relaxation of the Glides

The verb which contains among its radical letters a weak letter or more than one weak letter is called a Weak Verb. The weak letter comes either at the beginning, middle, or end of the verb. An example of verbs whose first radical letter is a weak letter is وعد The verb قال is an example of a verb which has a medial weak letter. An example of verbs with a final weak letter is دَنَوَ

Verbs which have an initial or medial weak letter undergo phonological variations but not orthographic. Such phonological variations have been considered in Alnajem (2002b). The orthographic variations emerge when the verb ends with a weak letter.

The last radical letter of the verb which ends with a weak letter can either be /w/ or /y/ (glide). This means that, at the lexical abstract level, a verb with a final weak letter ends with a glide. At the orthographic surface level, this glide may be changed to Extended Alef ألف ممدودة which is written as «ا» as in:

(12) دَعَوَ → دعا

or as Maqsoura Alef ألف مقصورة written as «ى» as in:

(13) رَمَيَ → رَمَى، ارتَمَيَ → ارتَمَى

A linguistically-motivated account for these variations has been given in Alnajem (2002c). The replace rules used to handle these variations are introduced also in Alnajem (2002c). Relaxation of the orthography of stem-final glides is achieved in the following way:

```
[ a a (–>) a y || – .#. , , a y (–>) a a || – .#. ] ;
```

This rule is in fact two rules which are applied in parallel. The special symbol ' , , ' is used to indicate the parallel application of two rules. Applying the two rules in sequence will do nothing since it will convert X to Y then convert Y to X. The first rule optionally changes the Extended Alef (a a) to Maqsoura Alef (a y) if the Extended Alef is at the end of the verb. This allows the system to accept verbs like دعا and دعى*. The second rule also optionally changes the Maqsoura Alef (a y) to Extended Alef (a a) if the Maqsoura Alef is at the end of a verb. This allows the system to accept verbs like رمى and رما*.

4.3.3 The Relaxation of Vocalization

In Arabic, words are written with three degrees of vocalization: full, partial, or no vocalization. For example, the following are three degrees of vocalization for the same verb:

(16)

a- قَتلت (full vocalization)

b- قَتَلت (partial vocalization)

c- قتلت (no vocalization)

Arabic text is normally written with no vocalization. Partial vocalization is used for disambiguation. For example, the verb قتل can be understood as an active voice form or a passive voice one. With partial vocalization, this ambiguity is eliminated. When we write the verb as قتل, we eliminate the passive form; while when we write it as قَتل, we eliminate the active form. Full vocalization is used in language teaching text books and in the Quran.

The different degrees of vocalization can simply be handled through this relaxation rule:

[V (–>) 0]

This rule says that a short vowel can optionally be deleted. The V is a defined variable which stands for the set of Arabic short vowels {a,u,i}. Beesley (1998) used a similar rule to deal with absent vocalization in his finite-state system.

5. Future Research

Future research will consider how the finite-state computational approach introduced in this paper can be expanded to design part of speech taggers and guessers. In addition, future research will consider how this computational approach can be used to design an Arabic spell checker. Future research will also cast light on using this finite-state approach to create morphological analysis/generation components for large-scale natural language processing systems like grammar checkers.

6. Conclusion

The complex variations existing in Arabic orthography represent a main cause of writing errors and confusion even among well-educated native speakers of Arabic. They also affect computer programs which deal with Arabic textual data like database management systems. The relaxation technique demonstrated in this paper is a special technique needed to allow computer systems to handle the errors caused by orthographic variations. This technique has been implemented in this paper through a finite-state computational approach.

Besides the negative aspects of the relaxation technique, the paper demonstrates some positive aspects of this technique. The paper shows that relaxation is a useful technique, and is needed in different fields, especially in the age of Information Technology. In this age, language users normally deal with large amounts of textual data, and they are required to enter their textual inputs to information systems (like e-mail client programs, web browsers, and database systems) at a maximum speed. This situation normally leads to writing errors which may have a harmful impact on language comprehension and/or the performance of information systems (like database systems).

Furthermore, the paper demonstrates how the finite-state technique can be used to handle orthographic variations, and to allow for the relaxation of such variations. Replace rules, which were used to handle orthographic variations (Alnajem, 2002c), have been used also to apply orthographic relaxation, but with the additional use of the optional rule application method.

WORKS CITED

AlDuqur, A.. Mu9jamu l-qawaa9idi l-9arabiyyati fi nnahwi wa ssarf. Damascus: Daaru l-qalam, 1986.

Alkhatib, A. ?usuulu l-?imlaa?. Damascus: Daru saadi ddiin, 1994

Alnajem,Salah. "Computational Approaches to Arabic Morphology." Diss. Essex University, 1998.

—. "A Finite-State Approach to Arabic Verbal Derivation." Unpublished essay. Kuwait University, 2002a.

—. "A Finite-State Approach to Arabic Verbal Inflection." Unpublished essay. Kuwait University, 20002b.

—. "A Computational Approach to the Variations in Arabic Verbal Orthography." Unpublished essay. Kuwait University, 2002c.

Assi, M. and Emil Ya'aqoub. ?al-mu ... jamu l-muffassalu fi l-lugati wa l-?adab. Beirut: Daaru l-9ilmi lil malaayiin, 1987.

Beesley, K. "Computer Analysis of Arabic Morphology: A Two-Level Approach with Detours." Perspectives on arabic Linguistics III: Papers from the Third Annual Symnposium on Arabic Linguistics. Eds. B. Comrie and M. Eid. Amsterdam: John Benjamin's Publishing Company, 1991. 155-72.

—. "Arabic Finite-State Morphological Analysis and Generation". COLING'96. Vol. 1. Copenhagen, 1996. 89-94.

—. “Arabic Morphology Using Only Finite-State Operations.” Computational Approaches to Semitic Languages: Proceedings of the Workshop. Ed. Michael Rosner. Montreal: University of Montreal, 1998. 50-57.

Bird, S and T. Ellison. One-Level Phonology: Autyosegmental Representations and Rules as Finite-State Automata. Technical report. Research paper EUCCS/RP-51. University of Edinburgh, 1992.

—. “One-Level Phonology”. Computational Linguistics. 20.1 (1994): 55-90.

Kay, M. “Nonconcatenative Finite-State Morphology.” Proceedings of the Third Conference of the European Chapter of the Association for Computational Linguistics. Copenhagen, 1987. 2-10.

Kiraq, George. “Multi-tape Two-level Morphology: A Case Study in Semitic Nonlinear Morphology”. Proceedings of Coling 94. 1994. 180-86.

—. *Computational Nonlinear Morphology: With Emphasis on semitic Languages*. Cambridge: Cambridge University Press, 2001.

Kornaiu, A. “Formal Phonology”. Diss. Stanford University, 1991.

Narayanan, A and L. Hashem. “On abstract finite-state morphology.” Proceedings of the Sixth Conference of the European chapter of the Association for Computational Linguistics, 1993. 297-304.

Roche, E. and Yves Schabes. “Introduction.” Finite-State Language Processing Eds. Roche, E. and Yves Schabes. Cambridge: MIT Press, 1997. 1-66.

NOTES

1. This initial glottal stop representation has a variant which is إ . This variant is used in some particles.
2. There are some minor roots whose last radical letter is considered as /w/ or /y/. In this case, the Alef can optionally be written as Extended or Maqsoura Alef as in
جثى/ جثا → جَثَي/ جَثَوَ
3. This word-initial glottal stop has a variant which is ‘إ’. This variant is used in specific articles.

* * *