International Journal of Information Science and Management
Vol. 13, No. 1, 2015, 39-50

Building a Syllabic Analyzer for Persian Using Finite State Transducers

Mohammad Amin Mahdavi
Department of Computer Engineering,
Imam Khomeini International University, Qazvin, Iran
mahdavi@researchattic.ca

Abstract

Persian follows a concatenative morphology, where new morphemes are generated by chaining different morphemes together to form a new compound word. Whenever, two morphemes bind to form a new morpheme, there is a possibility that the syllables at the morpheme boundaries undergo structural change. This study suggests that these syllabic alterations may be captured using a finite state approach. It further argues that syllabification may be incorporated into the process of lexicon building. This approach allows the syllabification rules to be encoded in the lexical knowledge, when a lexicon is built using the finite state methods. The rules captured here can also assist the processing of syllabic alterations in word boundaries as well. It is particularly useful to process meter in Persian poetry.

Keywords: Persian language, syllabic alterations, epenthesis, finite state morphology, lexicon, syllabification.

Introduction

Persian has abundant phonotactic rules to avoid phonemic oddities. Epenthesis in Persian, normally, occurs during the alterations. An affix binds to a morpheme to produce a new morpheme. At the boundary between the affix and the morpheme, phonological rules may require the insertion, deletion, or replacement of a letter. In finite state morphology, such rules are captured using replacement operators. At the morpheme boundaries in Persian, another interesting phenomenon takes place. The syllabic structures may also undergo alterations.

For instance, the singular form for the word "tree" in Persian is /*.de.raxt.*/ (درخت)[1]. The syllabic structure of the stem undergoes change when the plural suffix /*.ə̈ ān.*/ (أن) is affixed to the end. As a result, the plural /*.de.rax.tān.*/ (درختان) is formed. In this case, the elongated syllable /.raxt./ was shortened to /.rax./, causing /t/ to shift to the next syllable. Similarly, the syllabic structure in Persian does not allow for /*.tə̈ ān.*/ to occur, since there are two consonants in the initial position of the syllable. The phonetic rules of Persian require the deletion of the *hamza* (/*ӟ* /) leaving a single /t/ consonant in the initial position of the syllable.

The syllabic alterations in Persian can occur in combination with epenthesis, where additional consonants would have to be inserted at the morpheme boundaries. This study argues that all syllabic alterations in Persian can be captured using the finite state approach. It further argues that, if this knowledge is incorporated into a finite state lexicon, a finite state machine can be constructed to perform syllabification on Persian words.

Stating the motivation

Poetic composition in Persian follows a strict system of metering, which relies heavily on the syllabic arrangements in a verse. The essence of the poetic meter is captured by patterns of short and long syllables. The process of identifying the meter in a poem requires a scheme to translate the surface form into a sequence of short and long sounds known as the "u-dash" form. The "u-dash" notation refers to the representation of short and long syllables. Let us take the following half-line of poetry as an example.

verse	مرنجان دلم را که این مرغ وحشی maranjān delam rā keȟ ӟ īn morğe vaḩ şī										
Syllabification	ma	ran	jān	de	lam	rā	keȟ	ӟ īn	mor	ğe	Vaḩ	şī
Syllabic type	CV	CVC	CU	CV	CVC	CU	CV	CUC	CVC	CV	CVC	CU
u-dash notation	U	-	-	U	-	-	U	-	-	U	-	-

Figure 1. The process of translating a half-line of poetry to a sequence of u-dash notation.

The first step in the process of translating a half-line of poetry into the u-dash notation is to segment the surface form into its syllabic structure. The next step is to convert each syllable into its type, based on the length of the vowel and the number of consonants in the final position of the syllable. At the last stage, the syllabic types are translated into u-dash notations, "U" being a short syllable and "–" being a long syllable. In this example, the half-line follows a pattern of "short-long-long" repeated four times.

The motivation for this paper comes from automating this process. This paper tries to engage in the first step of the process, by offering a way to segment the words of a verse into their constituent syllabic structures.

Stating the problem

Short vowels are not explicitly written in Persian. The author is of the belief that vowels may be considered as diacritical marks for vocalization purposes. The absence of short vowel poses a serious challenge for any syllabification effort. This is because vowels are the core components of a syllable. Every syllable, in Persian, is recognized by its single vowel (Dehghan & Kord-e Zafaranlu Kambuziya, 2012). While long syllables are always written explicitly, short vowels are often omitted. Omission of vowels from a word makes syllabification an ambiguous process.

It is argued in this paper that, if the syllabic alteration rules are incorporated into the lexicon building process using a finite state approach, the resulting transducer is able to suggest the proper syllabification for a given word. The lexicon, which is in mind here, is based on a two-level morphology concept.

Outlining a solution

The argument here is that, if the syllabification is incorporated into the morpheme structures, the morphotactics will automatically carry the syllabic information throughout the process. This paper demonstrates that at morpheme boundaries, the syllabic structure may undergo alteration. These alterations can be captured using cascading re-write rules.

In other words, instead of building a lexicon that breaks a word into its morphological structure, it is possible to build a transducer that breaks a word into its constituent syllables. However, the process of building this transducer is inspired by the process of building a lexical analyzer. The inspiration comes from the notion that both of these processes use morphotactic knowledge to build the transducer. The difference between a lexical analyzer and a syllabic analyzer is in the information captured by the upper level. A syllabic analyzer encapsulates the syllabic structure in the upper level, while a lexical analyzer captures that morphological structure and part of speech meta-data.

The premises

This study is based on two premises 1) phonological structure of Persian syllables may be captured by orthography 2) syllabic alterations in Persian can only happen at the morpheme boundaries.

Syllabification is a task that acts on the phonological structure of a word. The assumption in this paper is that Persian phonological structures may be captured using orthographic representations. While, phonologically, it is possible to start a word with a vowel, a standard orthographic representation would require the insertion of an intervening consonant to prevent such occurrences. Thus, it is important to begin the process with a fully vocalized and standard orthographic representation of morpheme that correctly captures the phonological features of the language.

One may envisage an intermediate orthographic representation that captures the phonological features as well as the syllabic structure. For the purposes of this study, the syllabic analyzer (i.e. the transducer) will have the surface form of the word at the lower level, while capturing the orthographic representation of phonology and syllabic structure at the upper level. For instance, “انگور” (/*angur*/ meaning grape) in the lower level is mapped to /*.ǟ an.gūr.*/ in the upper level.

The second premise in this study states that structural changes in syllables only occur at the morpheme boundaries. This indicates that, it is only during the word formation process that syllabic structures undergo change. Therefore, it is possible to compute the syllabification, if one were to use morphotactics to form a word. Much of this paper focuses on the rules for syllabic alteration at the morpheme boundaries.

Two level morphology

Since its introduction by Koskenniemi (1983) and Kaplan and Kay (1994), finite state technology is deemed suitable for describing the morphological phenomena among concatenative languages. A few tools have been developed to accommodate the compilation of extended regular expression into finite state automata (FSA) and finite state transducers (FST) (Karttunen, Chanod, Grefenstette, & Schiller, 1996; Mohri, 1996; van Noord & Gerdemann, 2001). At the core of the two level morphology lies the idea that transducers possess the ability to capture the linguistic alterations. Finite state transducers are machines that accept a sequence of symbols as input and generate another sequence of symbols in form of an output. In other words, an FST can accept a surface form at one level and produce an analysis on another level. One of the distinguishing features of finite state transducers is that they operate on two levels.

Persian is a language that undergoes phonological as well as orthographic alterations at different stages of morpheme construction. The general claim of finite state morphology is that, in addition to morphological rules, alteration phenomena may be compiled into finite state transducers.

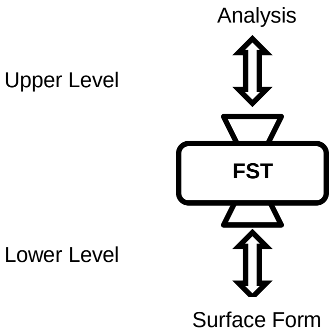


Figure 2: The general architecture for a finite state transducer.

Syllabic alterations at morpheme boundaries in Persian may consist of assimilation, deletion, and epenthesis among others. Such alterations are commonly captured by "rewrite rules". In late 1960s, Chomsky and Halle (1968) formalized a grammar for phonological alterations. They suggested a sequence of ordered rewrite rules in form $\boldsymbol{\alpha \rightarrow \beta / \varphi \,\_\_\, \omega}$, where **α, β, φ**, and **ω** may be any complex strings, and the "__" notation signifies the context in which **α** occurs. This rule may be interpreted as, "**α** may be replaced by **β**, if it occurs **φ**, and before **ω**." These rules are also known as the *context-sensitive rewrite rules*. According to Chomsky's grammar classifications, these rules are more powerful in their descriptive power than regular expressions (RE) and context-free grammar (CFG). However, Johnson (1972) observed that such rules are not as powerful and may be captured by a finite state transducer. At a later stage, Kaplan and Kay reformulated such rewrite rules as regular relations and demonstrated that they may be represented by a finite state transducer (Kaplan & Kay, 1981; Kaplan & Kay, 1994). Beesley and Karttunen (2003) have incorporated these rules into practical applications for finite state morphology.

Semantically, the rewrite rule $\alpha \rightarrow \beta / \varphi \_\_ \omega$ can be interpreted as: 'any arbitrary string α may be replaced by another arbitrary string β, when in it occurs in the context of φ on the left and ω on the right'. Kaplan and Kay (1994) followed Johnson's idea that, each linguistic feature that is captured by a rewrite rule can be represented by a single transducer. They showed that cascading several transducers using the composition operator produces a single transducer that represents all intermediate alterations.

Syllabic structures in Persian

Syllabic structure in Persian is comprised of three positions; 1) initial (onset), 2) medial (Nucleus), and 3) final (coda)[2]. The initial position is always filled with a single consonant. Other source (Dehghan & Kord-e Zafaranlu Kambuziya, 2012) have reported a null for the initial position. However, the author is of opinion that such cases only occur in the first syllables of a word and a *Hamza* is always implicitly present in the initial position to make it audible. The medial position is always filled with a vowel. The final position, however, may remain unfilled. It may also be filled by one or two consonants. Therefore, every string in Persian has as many syllables as there are vowels in it. This is because vowels can only occur in one position within the syllable. The rest of the positions are filled with consonants. Conversely, Persian has six distinct templates for its syllabic structures.

Table 1

Every syllable begins with a single consonant followed by either a single short vowel or a single long vowel followed by zero, one, or two consonants. C = consonant; V = Short vowel; V□ = Long vowel.

Initial position	**Medial position**	**Final position**	
C	V		
C	V□		
C	V	C	
C	V□	C	
C	V	C	C
C	V□	C	C

Persian has three short vowels (i.e. /a/, /e/, and /o/) and three long ones (i.e. /ā/, /ī/, and

/ū/). The remaining letters in Persian are all consonants. Unlike English, where multiple vowel such as /au/ and /oo/ occur, Persian does not allow the occurrence of multiple vowels in a single syllable. The syllabic structure in Persian, also, prohibits the clustering of multiple consonants in the initial position. The only instance, where two consonant are maximally clustered together in Persian syllables, is in the final position. As mentioned in section 5, it is phonologically possible to start a morpheme with a vowel. However, using an intermediate orthographic representation, an intervening consonant (i.e. glottal *hamza*) is inserted at the beginning of the morpheme to prevent a syllable to begin with a vowel.

Therefore, any catenation of bordering syllables, which causes these rules to be broken, may result in syllabic structure change. In the following sections, an overview of phonological features and alterations are introduced.

Role of glottal stops in syllabic alteration

Persian alphabet has two glottal stops, *hamza* (/ء/) and *'ayn* (/ع/). The former has a peculiar behavior, when it occurs in the initial syllabic position. It seems that *hamza* in the initial syllabic position acts as a seat for the following vowel and does not possess the full consonant status. In such cases, *hamza* is treated as a weak consonant; more like a placeholder.

This unusual characteristic of *hamza* gives rise to a simulated hiatus, when it follows a syllable that ends with a vowel. Whenever a syllable ending in a vowel is followed by another syllable that begins with a glottal *hamza*, the *hamza* would have to alter to avoid the clustering of two vowels.

.se.pā.hī.	**+**	**.ӭ ān.**	**→**	**.se.pā.hī.yān.**
Soldier		*Suffix*		*Soldiers*
Noun+Sg		*Plural*		*Noun+Pl*

Example 1: here, the weak glottal *hamza* (/ӭ /) is replaced by an intervening consonant /y/ to prevent the long vowel /ī/ and long vowel /ā/ to co-occur next to each other.

The initial glottal *hamza* in Persian syllables has one more characteristic. When *hamza* follows another syllable that ends in a consonant, the ending consonant of the previous syllable is shifted to replace the glottal *hamza* in the following syllable.

.ba.rā.dar.	**+**	**.ӭ ān.**	**→**	**.ba.rā.da.rān.**
Brother		*Suffix*		*Brothers*
Noun+Sg		*Plural*		*Noun+Pl*

Example 2: here, the last consonant /r/ is shifted to the next syllable and replaces the weak *hamza* (/ӭ /) to ensure that the initial position of the syllable is filled with a consonant.

Role of silent /*ȟ*/ in syllabic alteration

Every word in Persian must either end in a consonant or a long vowel. Words must not end in a short vowel, save the *izafeh* constructs. This phonetic restriction has given rise to a silent /*ȟ*/ that is never pronounced. Invariably, the silent /*ȟ*/ is added to words that end in a short vowel. For instance, the word for school (e.g./*.mad.re.seȟ.*/, pronounced as /*.mad.re.se.*/) has a silent */ȟ/* at the end to prevent the word from ending in short vowel /e/.

The silent /*ȟ*/ is normally occurred at the end of atomic morphemes (i.e. the root). A silent /*ȟ*/ can only occur in the middle of a word, if one or more suffixes are attached to the end. However, a silent /*ȟ*/ may only be retained, if the following suffix does not begin with a glottal *hamza*. A silent /*ȟ*/ may also be altered to a different consonant.

ج

.beȟ .	+	**.ӛ eş.**	→	**.be.heş.**
'To' *preposition*		*Personal* *Pronoun* *Suffix*		*To him/her*

Example 3: in this example, the silent /ȟ / is contracted to a voiced /h/ and is shifted to replace the glottal *hamza* in the next syllable.

.xā.neȟ .	+	**.hā.**	→	**.xā.neȟ .hā.**
house		*plural* *Suffix*		*houses*

Example 4: In this example, the silent /ȟ / is retained, since the following syllable starts with a consonant.

Epenthesis in Persian

Epenthesis in Persian only occurs in morpheme boundaries. Epenthesis always occurs between adjacent syllables at the morpheme boundaries. The syllable structure in Persian is such that a vowel may not be added or removed from a syllable. Thus, the only form of epenthesis allowed in Persian is the alteration of consonants. As demonstrated in the previous section, in addition to intervening consonants, shifting of a consonant to the adjacent syllable may also be coupled with the epenthesis.

In terms of syllabic neighborhood, two types of adjacency would have to be considered. The first case is when a prefix is attached to the following morpheme. The second type of adjacency is when a stem attaches to a suffix. In the former adjacency, a prefix (the left context) may end in a short vowel. However, this is not possible for a stem or a free morpheme to end in a short vowel.

PREFIX STEM .be. .ẍ ān. a) In this case, a prefix may end in a short vowel.	STEM SUFFIX .nā.meȟ . .hā. b) In this case, the stem may never end in a short vowel.

Figure 3: In Persian, morpheme boundaries may be one of the two types. The syllable structure for the left context would depend on the type of morpheme boundary. For a suffix boundary, the left contex cannot end in a short vowel. However, in a prefix boundary, the left context may end in a short vowel.

In terms of the syllabic structure, it is possible to characterize the preceding syllable at a morpheme boundary according to the following templates.

Preceding syllable

initial	medial	final	
(onset)	(nucleus)	(coda)	
C	V	(Ȟ)	
C	V	C	(C)
C	V̄	(C)	(C)

Figure 4: The left context of a morpheme boundary may have all of the typical syllable structure. However, in cases where the left context is not a prefix, it is not possible to have the CV form. Instead, a silent /Ȟ/ may fill the consonant in the coda position. (C) means the consonant at that position is optional.

The right context also has peculiarities that revolves around the initial consonant. When the onset of the right context is filled with a weak glottal *hamza*, it is highly likely that an epenthesis would occur. The template for the onset position being filled with various consonants is captured as follows.

Proceeding syllable

initial	medial	final	
(onset)	(nucleus)	(coda)	
C	V	(C)	(C)
Ӭ	V	(C)	(C)

Figure 5: The right context of a morpheme boundary may have all of the typical syllable structure. However, the initial consonant position may be filled by a weak glottal *hamza* (/Ӭ /). It is this glottal *hamza* that may cause an alteration in the syllabic structure of the morpheme boundary. (C) means the consonant at that position is optional.

Two level rules for syllabic alterations in Persian

This paper argues that Persian syllabic alterations may be incorporated into the construction of transducer using the context sensitive rewrite rules. To demonstrate this, an example is provided to elaborate on the process. Figure 6 describes the general morphotactic rules in Persian.

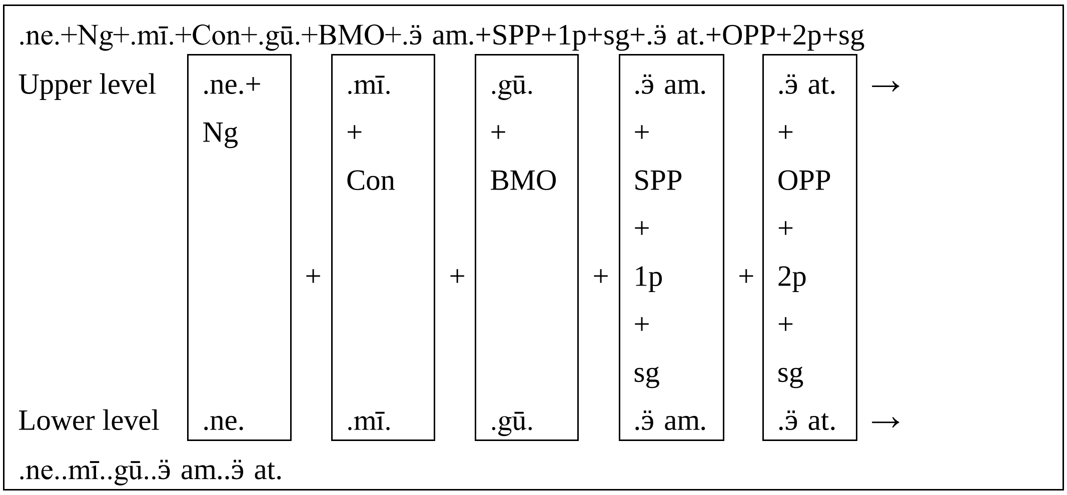


Figure 6: The general morphotactics of Persian is captured by a regular relation. The regular relations can be represented by a finite state transducer. Here, Ng (negative), Con (continuous), BMO (bon mozare'), SPP (subjectival personal pronoun), 1p (first person), sg (singular), OPP (objectival personal pronoun), and 2p (second person) are POS tags.

Direct catenation in Persian does not yield new morphemes. A sequence of alteration rules will have to be applied to the lower side of the transducer using the composition operator, before the surface form is yielded.

Similarly, the upper level of the transducer in Figure 6 will have to be cascaded with a series of rewrite rules to apply the syllabic alterations at the morpheme boundary. As established in previous sections, syllabic alterations in Persian occur at the morpheme boundaries. Syllabic structures that occur in the suffix position and begin with a weak consonant (glottal *hamza*) may trigger an alteration. Similarly, prefixes that end with a short vowel are likely to cause a syllabic alteration.

Figure 7 demonstrates the application of syllabic alteration rules. Each rewrite rule represents one syllabic alteration. When the three rules are cascaded using a composition operator, the resulting transducer will have incorporated all the necessary alterations to achieve the correct syllabification.

Upper level		**Context-sensitive rewrite rule**		**Lower level**
.ne..mī..gū..ə̈ am..ə̈ at.	→	ə̈ → y ‖ {ū..} __ a	→	.ne..mī..gū..yam..ə̈ at.
.o.				
.ne..mī..gū..yam..ə̈ at.	→	[..] → {..} ‖ V __ C {..ə̈ }	→	.ne..mī..gū..ya..m..ə̈ at.
.o.				
.ne..mī..gū..ya..m..ə̈ at.	→	{..ə̈ } → 0 ‖ {.}C __ V	→	.ne..mī..gū..ya..mat.

Figure 7: A cascade of alteration rules is applied to the lower levels of the FST in Figure 5 using the composition operator.

As it can be seen above, three consecutive applications of syllabic alteration rules have altered the lower string from Figure 6 to the final and correct syllabification format. Here, the alterations are written using context-sensitive rewrite rules. These rewrite rules are compiled into finite state transducers using FOMA syntax (Hulden, Foma: a finite-state compiler and library, 2009). FOMA is an open source regular expression compiler that has a syntax similar to that of XFST by Xerox®.

Producing a syllabification transducer

Once the construction of the finite state transducer is complete according to the stages identified in this paper, two separate products may yield. The first product is the finished lexicon. As it can be observed, the upper level of the transducer contains the morphological information and the lower level has the syllabified surface form. The final stage of completing the lexicon is to apply one last rewrite rule to the lower level of the FST in Figure 7. This last rewrite rule would replace the syllabic marker (i.e. the full stop) with a null symbol; hence completing the surface structure.

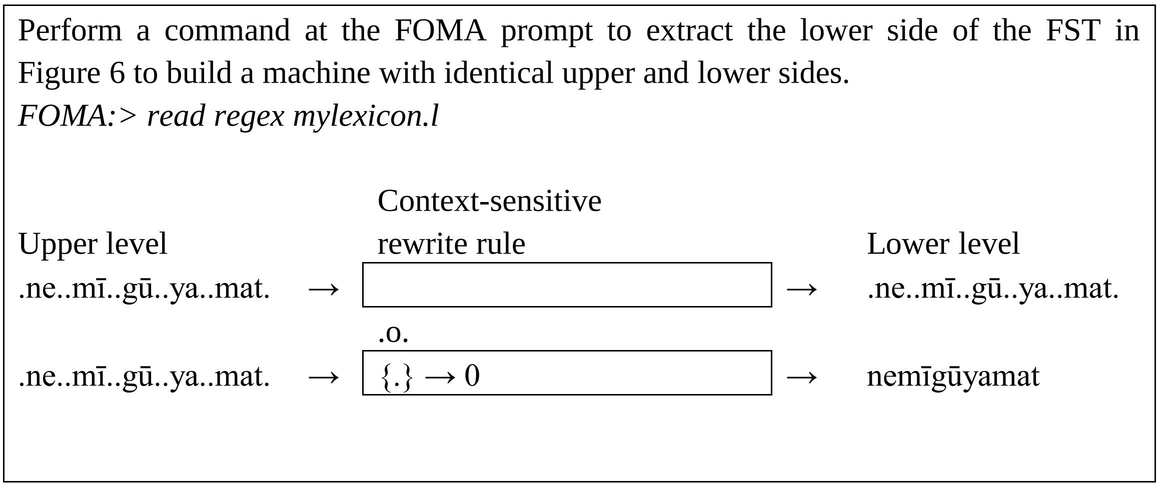
Upper level		**Context-sensitive rewrite rule**		**Lower level**
.ne..mī..gū..ya..mat.	→	{.} → 0	→	nemīgūyamat

Figure 8: The last stage in finite state lexicon construction. It replaces all syllabic markers into a null symbol.

However, given the transducer in Figure 7, it is possible to extract the lower side of the machine and turn it into another transducer that has identical upper and lower side. As observed in Figure 9, using a FOMA command, the lower side of the transducer generated in

Figure 7 is extracted into an independent transducer, whose upper and lower sides are identical. The lower side of the new transducer contains syllable marks indicating syllable boundaries in the word. Of course, two consecutive syllable marks indicate a morpheme boundary in the word.

Since the upper and lower sides have identical languages, it is possible to compose this machine with another FST that would eliminate the syllable mark from the lower side. The yielded machine would take the surface structure of a word in Persian and return the syllabification of that word from the upper side. Similarly, the upper side can also be cascaded with a rewrite rule that converts two consecutive syllable marks into one. The resulting transducer is a syllabic analyzer that contains the surface form at the lower side and the syllabic structure as the upper side.



Perform a command at the FOMA prompt to extract the lower side of the FST in Figure 6 to build a machine with identical upper and lower sides.

FOMA:> read regex mylexicon.l

Context-sensitive rewrite rule
Upper level
Lower level
.ne..mī..gū..ya..mat. → → .ne..mī..gū..ya..mat.
.o.
.ne..mī..gū..ya..mat. → {.} → 0 → nemīgūyamat

Figure 9: A two-step process to build a syllabification machine.

Conclusion

Syllabification of Persian is a challenging task. This is due to omission of short vowels in the written text. It is possible to design a finite state transducer to enable the automation of the syllabification task. This paper proposes to build the transducer with syllabic alteration rules in mind. In addition to morphotactic rules, syllabic alteration rules are incorporated in the process. The final product is a transducer that maintains syllabic knowledge. In a post processing stage, a transducer yields and accepts the surface structure of a Persian word and returns the syllabic structure of the word.

Endnotes

1. For a full treatment of the transliteration style, please see (Mahdavi, 2012) at http://ijism.ricest.ac.ir/ojs/index.php/ijism/article/view/129
2. For more, please see (Hulden, Finite-State Syllabification, 2005) and (Dehghan & Kord-e Zafaranlu Kambuziya, 2012)

References

Beesley, K., & Karttunen, L. (2003). *Finite State Morphology.* Stanford: CSLI.

Chomsky, N., & Halle, M. (1968). *The Sound Pattern of English.* NewYork: Harper and Row.

Dehghan, M., & Kord-e Zafaranlu Kambuziya, A. (2012, January). A Short Analysis of Insertion in Persian. *Theory and Practice in Language Studies, 2*(1), 14-23.

Hulden, M. (2005). Finite-State Syllabification. *FSMNLP, volume 4002 of Lecture Notes in Computer Science* (pp. 86-96). Springer.

Hulden, M. (2009). Foma: a finite-state compiler and library. *EACL '09 Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics: Demonstrations Session* (pp. 29-32). Stroudsburg: Association for Computational Linguistics.

Johnson, C. D. (1972). *Formal Aspects of Phonological Description.* The Hague: Mouton.

Kaplan, R. M., & Kay, M. (1981). Phonological rules and finite-state transducers. *Linguistic Society of America Meeting Handbook; Fifty-Sixth Annual Meeting.* New York: Linguistic Society of America.

Kaplan, R. M., & Kay, M. (1994). Regular models of phonological rule systems. *Computational Linguistics, 20,* 331-378.

Karttunen, L., Chanod, J. P., Grefenstette, G., & Schiller, A. (1996). Regular expressions for language engineering. *Natural Language Engineering, 2*(4), 305 - 328.

Koskenniemi, K. (1983). *Two-Level Morphology: a General Computational Model for Word-Form Recognition and Production.* Helsinki: The Department of General Linguistics, University of Helsinki.

Mahdavi, M. A. (2012). A Proposed UNICODE-Based Extended Romanization System for Persian Texts. *International Journal of Information Science and Management (IJISM), 10*(1), 57-71.

Mohri, M. (1996). On some applications of finite-state automata theory to natural language processing. *Natural Language Engineering, 2*(1), 61 - 80.

van Noord, G., & Gerdemann, D. (2001). An extendible regular expression compiler for finite-state approaches in natural language processing. In O. Boldt, & H. Jurgensen (Ed.), *Automata Implementation, 4th International Workshop on Implementing Automata.* Germany: Springer.