Finite-State Syllabification

Mans Hulden

The University of Arizona
Department of Linguistics
PO BOX 210028
Tucson AZ, 85721-0028
USA
mhulden@email.arizona.edu

Abstract. We explore general strategies for finite-state syllabification and describe a specific implementation of a wide-coverage syllabifier for English, as well as outline methods to implement differing ideas encountered in the phonological literature about the English syllable. The syllable is a central phonological unit to which many allophonic variations are sensitive. How a word is syllabified is a non-trivial problem and reliable methods are useful in computational systems that deal with non-orthographic representations of language, for instance phonological research, text-to-speech systems, and speech recognition. The construction strategies for producing syllabifying transducers outlined here are not theory-specific and should be applicable to generalizations made within most phonological frameworks.

1 The Syllable[1]

Phonological alternations are often expressed efficiently by reference to syllables. Most phonological descriptions presume a regular grouping of C or V elements into syllables which other phonological rules can subsequently refer to.

An example of syllables being used as a domain of phonological alternations is given by Kahn [1], who noted that an underlying [t] phoneme in English may behave in various different ways, conditioned mainly by its position in the syllable. A [t] can surface:

- as aspirated [t$^{\text{h}}$], as in **crea<u>t</u>ivity**
- as glottalized [t$^{\text{ʔ}}$], as in **crea<u>t</u>e**
- as [t], as in **s<u>t</u>em**
- as a flap [ɾ], as in **crea<u>t</u>ing**
- as [č$^{\text{h}}$], as in **<u>t</u>rain**
- as [č], as in **s<u>t</u>rong**

Many other phenomena are sensitive to syllable boundaries. A further example would be, for instance, syncope (schwa-deletion) where words like **licorice** may surface either as [lɪ.kə.rɪš] or as syncopated [lɪk.rɪš], as noted by Hooper [2].

[1] Thanks to Mike Hammond, Lauri Karttunen, and two anonymous reviewers for guidance, comment, and discussion. Any errors are my own.

A. Yli-Jyrä, L. Karttunen, and J. Karhumäki (Eds.): FSMNLP 2005, LNAI 4002, pp. 86–96, 2006.

More abstract levels of representation in phonological theory—such as metrical systems in which the structure involves the laying down of feet—also assume the existence of syllables at some lower level.

To make accurate predictions about syllabification, both in phonological behavior and in empirically attested preferences, requires—as in the case of [t] mentioned above—subtle differentiation of syllabification patterns with respect to consonant cluster affiliation. We present an approach based on a fairly traditional view of the syllable that largely follows the sonority hierarchy and the maximum onset principle. Knowledge of word stress is not assumed in the syllabifier—cases where word stress appears to affect syllabification have been modelled by sensitivity to the quality of syllabic nuclei and of the surrounding consonant clusters.

Table 1. Regular expression operators

A*	Kleene star
A+	Kleene plus
A | B	Union
(A)	Optionality, equivalent to A|0
~A	The complement of A
A B	Concatenation
A.l	Extraction of the lower language in relation A (the range of A)
A.u	Extraction of the upper language in relation A (the domain of A)
A .o. B	Composition
A .P. B	Upper-side priority union, equal to A | [~[A.u] .o. B]
A -> B ‖ L _ R	Directed replacement with context restriction
A @-> B ‖ L _ R	Left-to-right longest replace with context restriction
A @> B ‖ L _ R	Left-to-right shortest replace with context restriction
A -> B . . . C	Left-to-right marking operator with context restriction

2 Finite-State Syllabification Methods

The finite-state formalism owes much of its conceptual background to phonological rewrite systems originating in the Sound Pattern of English [3]. Kaplan and Kay [4] subsequently provided a strong connection between classical generative phonology and finite-state systems. The syllable, however, had no official recognition in much of the early generative work, and when it later entered into the scope of research, a rich internal structure of the syllable was assumed to the extent that syllabification processes were no longer commonly described with rewrite rules—although verbal descriptions of syllabification "algorithms" were often given.

The finite-state calculus rewrite operators (see table 1) provide most of the functionality required for a convenient description of most details in syllabification processes.[2] Depending on the complexity of a language's syllables, syllabifiers may need to have refined knowledge of the types or quality of

[2] The description here assumes the Xerox xfst formalism [5].

phonemes—consonants in particular. Finnish, as an example of a language with a relatively simple syllabification process, can be treated with little regard to consonant clusters:[3]

```
C* V+ C* @-> ... "." || _ C V
```

However, languages such as English that feature a variety of syllable types will need to be treated with detailed attention to the quality and order of segments.

For designing the syllabifier described here, the syllabifications of 1,920 words that all contained consonant clusters were extracted from Merriam-Webster's Collegiate Dictionary and used as a set of empirical data to compare against.[4] Barring internal inconsistencies, the final predictions made by the syllabifier agreed with the source.[5]

3 Sonority

Languages that contain complex clusters of consonants are usually guided in their syllable structure by the concept of a sonority hierarchy. The principle states that more "sonorous" elements appear closer to the syllable nucleus, which in turn is the most sonorous element. The onset of a syllable thus mirrors the coda in sonority.[6]

Table 2. The sonority hierarchy

Increasing → sonority					
Voiceless Obstruents	Voiced Obstruents	Nasals	Liquids	Glides	Vowels
p,t,k,s,..	b,d,g,z...	m,n,ŋ	l,r,..	y,w,...	a,e,o,u...

[3] It is assumed that the legal vowels and consonants are defined in the sublanguages C, V. This treatment requires some further elaboration about legal dipthongs. The syllabification here is the traditional treatment [6]. It may be argued that the Finnish syllable is subject to additional sonority constraints—the rewrite rule here would yield /abstrakti/ → /abst .rak.ti/, whereas most native speakers prefer /abs.trak.ti/ or /ab.strak.ti/. Insofar as the syllable is permitted independent status as an entity outside language-internal phonological processes, accurate modelling of even Finnish, which has a relatively poor syllable inventory, is probably best treated in the manner outlined in this paper.

[4] http://www.britannica.com/dictionary

[5] In some cases the dictionary showed conflicting syllabifications for highly similar words. For instance, the words **poster**, **toaster**, and **coaster** were syllabified [pos.tər], [to.stər], and [kos.tər], respectively. The majority account was followed whenever the data were inconsistent. In this case, it was concluded that [(p|t|k)os.tər] would be the preferred syllabification.

[6] This observation is often attributed to O. Jespersen, *Phonetische Grundfragen* (1904).

In English, the word **comptroller**, for example, has a four-consonant medial cluster. This will be divided by the sonority sequence requirement into **mp.tr**.

English, by and large, adheres to the sonority requirements, with the exception of [s] which (in this treatment) must occur syllable-initially or syllable-finally (in word-medial position) and [h], which only occurs syllable-initially, never syllable-finally.

From a finite-state point of view, the sonority hierarchy is a statement dictating a particular order in which elements must occur in a legal syllable. The requirements of sonority are, however, not sufficient to syllabify correctly—an approach that only followed sonority requirements will massively overgenerate (see table 3):

```
define Onset [(VLObs) (VObs) (Nas) (Liq) (Gli)];
define Coda [(Gli) (Liq) (Nas) (VObs) (VLObs)];
define Syllable [Onset Vow Coda];
define Syllabify [Syllable -> ... "." || _ Syllable];
```

Here, we define the syllable to consist of onsets and codas, which are mirror images of each other according to the sonority hierarchy. We then introduce syllable boundaries between all legal syllables.

Table 3. Syllabifying by only sonority

/æbrəkədæbrə/	/kæləforniə/
æb.rək.əd.æb.rə	kæ.ləf.or.ni.ə
æb.rək.əd.æ.brə	kæ.ləf.orn.i.ə
æb.rək.ə.dæb.rə	**kæ.lə.for.ni.ə**
æb.rək.ə.dæ.brə	kæ.lə.forn.i.ə
æb.rə.kəd.æb.rə	kæl.əf.or.ni.ə
æb.rə.kəd.æ.brə	kæl.əf.orn.i.ə
æb.rə.kə.dæb.rə	kæl.ə.for.ni.ə
æb.rə.kə.dæ.brə	kæl.ə.forn.i.ə
æ.brək.əd.æb.rə	
æ.brək.əd.æ.brə	
æ.brək.ə.dæb.rə	
æ.brək.ə.dæ.brə	
æ.brə.kəd.æb.rə	
æ.brə.kəd.æ.brə	
æ.brə.kə.dæb.rə	
æ.brə.kə.dæ.brə	

3.1 Sonority Distance

Phonological theory also makes use of the concept of sonority distance, which states that consecutive sounds within a syllable must be sufficiently distant from

each other in terms of sonority [7]. The exact requirements vary from language to language: in English, [p] (a stop) may not be followed by an [n] (a nasal), although this is possible in e.g. French.

4 Maximum Onset

Another generalization about syllabification processes is that, given a choice between affiliating a consonant to a coda or to an onset, affiliating with the onset is preferable, cf. [1, 8].

Application of this principle can be used to eliminate overgeneration, and immediately narrows down the eligible syllabifications to a single one, i.e. [æbrəkədæbrə] → [æ.brə.kə.dæ.brə].

The combination of sonority requirements and onset maximization can be economically expressed through the shortest replace operator [5], assuming we have a definition of allowed onsets and coda clusters.

```
define Syllable Onset Vow Coda;
define MainRule Syllable @> ... "." || _ Syllable;
```

Table 4. Legal two consonant onsets in English. The obstruents are not quite symmetrical with respect to the consonants that are allowed to follow them. The phonemes {y,r} behave more alike than for instance the natural grouping of glides, {w,y}. This is also true for three-consonant onsets. Circles mark clusters that are legal only word-initially, and thus not included in the grammar.

	Gli		Liq		Nas		Sto		
	w	y	r	l	m	n	p	t	k
p			●	●					
t	●	●	●						
k	●	●	●	●					
b			●	●					
d	●	●	●						
g	●	●	●	●					
f			●	●					
θ	●	●	●						
š			○						
s	●	●		●	●	○	●	●	●

Table 5. Three consonant onsets in English

	w	y	r	l	m	n
sp		●	●	●		
st		●	●			
sk	●	●	●	○		

The shortest replace operator @> works like the standard replace operator, but will construct a transducer that follows a strategy such that the application site of the left hand side of the rule will be kept to a mimimum if there are alternative ways of applying the rule (i.e. if there are several legal ways to distribute the syllable boundary at the coda-onset juncture). Technically, this minimizes the coda instead of maximizing the onset, but the end result is equivalent. See tables 4 and 5 for the particulars of allowed onsets and codas in the English implementation here.

5 Stress

Many treatments of the English syllable found in the literature also depend on knowledge of stress. The generalization is that at least some consonants, [s] and the nasals in particular, tend to affiliate with a stressed syllable, going against the Onset Maximization principle. In the M-W data used for this implementation, some pairs where this is seen include [æm.yʊ.lɛt] vs. [ə.myuz] and [æs.pɛkt] vs. [ə.spɛ.rə.ɾi].

In this treatment, the goal has been to give an account of English syllabification without knowledge about the particular stress of a word, but based on the quality of vowels and surrounding consonant clusters. Still, most speakers of English do have a strong intuition about consonants affiliating to a coda in some syllables based on what appear to be stress factors. So, for instance, there is a tendency to syllabify **astir** as [ə.stɻ̩], but the proper name **Astor**, as [æs.tɻ̩].

To solve this without relying on knowledge of word stress, we have modeled consonant affiliation by adding two rules where nasals and [s] affiliate to the left when preceded by an open syllable where the nucleus is not {ə,i} to give the desired predictions.[7] These rules apply before the main syllabification rule:

```
define sRule[s -> ..."." || ([[Cons]|[(Stop) r]]) [Vow - ə - i] _ Cons+ Vow]];
define NasRule [Nas -> ..."." || [Vow - ə - i] _ y ];
```

6 Medial vs. Marginal Clusters

Often the types of onset that are found word-initially can be used as clues to deduce further restrictions on top of the sonority considerations [9]. As English allows, for instance, initial [spr] in many words (spring, spray, etc.), the conclusion can be drawn that [spr] should be legal in word-medial onsets as well. However, in modeling the syllabifications of a particular source (M-W), it has become clear that there is a tendency to avoid generalizing from some attested

[7] The syllabifier described was designed to be used as part of research concerning generalizations about English stress where an underlying representation was assumed that was close to the phonetic form of the word. Part of this research involved the separation of syllabification and stress rules, where syllabification would apply first, and stress later, and where the two would function as independent processes.

word-initial onsets to legal medial onsets. Although [sn] is a cluster very commonly encountered word-initially, as in e.g. **snow**, allowing the same cluster in word-medial position will not yield correct syllabifications in words such as **pilsner**, which, if [sn] were permitted, would be incorrectly syllabified as [pɪl.snɻ].[8]

Thus, certain initial clusters can probably not be used as a basis for legitimizing medial clusters of the same type. The initial-cluster [skl], for instance (which only occurs in a handful of words: sclerosis, sclaff, etc.), is one that has not been permitted syllable-initially in the syllabifier. Similarly with final clusters, e.g. [sɪksθs] is a unique and highly marked four-consonant cluster and does not seem to warrant the inference that [ksθs] would be a legal coda. For such coda clusters, this is in most cases not significant because of the tendency to maximize onsets—long codas will rarely be allowed except word-finally. In fact, the set of permitted codas have been modelled simply as any maximally two-consonant combination.[9] This makes exactly the same predictions as a model where codas are constrained to actually attested ones.

Onsets, on the other hand, must be attended to in more detail than the guiding sonority principles. In this implementation we have only marked syllable *boundaries*. In such a process, the main syllabification rule (above) applied to a word with an initial [skl]-cluster will never match [skl] since it is not a legal onset. But as the input language to the transducer is the universal language ?*, [s] will be transduced to [s], and [kl] will be matched as a legal onset as the syllabification proceeds. In effect, the initial [s] will be treated as "extrametrical."

Incidentally, the exclusion of onset clusters such as [skl] yields different syllabifications for word pairs such as **exclaim** and **explain** ([ɪks.klem], [ɪk.splen]).[10]

This strategy will not affect the final syllabification as long as we are content with marking syllable boundaries, not beginnings and endings. Such an approach should be sufficient for most applications since any phonological rule that later needs to refer to a syllable boundary in its conditioning environment will not need to know whether the boundary marks the beginning or the end of syllable.

If we wanted to "wrap" every syllable with both a beginning and end marker, $[_\sigma$ and $]_\sigma$, this issue would have to be addressed. However, we know of no simple phonological process in English that would require a differentiation between $[_\sigma$ and $]_\sigma$.

It should be noted that this implementation assumes an underlying form that is very close to the phonetic form. Applications that make use of more abstract underlying forms can derive further predictions through wrapping

[8] The discrepancy between acceptable word-medial and word-marginal syllable types has been the subject of much recent research. For a stochastic perspective, see Coleman and Pierrehumbert [10], and for an OT-related analysis, see Hammond [9].

[9] That this approach works has an interesting parallel in the OT literature, where a constraint with a similar function, such as ALIGN-3μ, is sometimes seen [11]. This constraint prohibits syllables heavier than 3 moras, except word-finally. For English, the prediction is quite similar to disallowing more than two coda consonants.

[10] M-W has this syllabification. This example pair 1) [ɪks.klem] and 2) [ɪk.splen] would indirectly make the subtle prediction that the [k] is aspirated [k^h] in 1), whereas the [p] would remain unaspirated in 2).

syllables with beginning and end markers instead of simply marking boundaries. For instance, the phonological phenomenon of Stray Erasure [12], where coda segments that cannot be legally parsed into syllables remain unpronounced, could be described by wrapping syllables. Supposing the underlying form of a word such as **damn** were [dæmn], instead of [dæm], as here, and supposing syllables would be grouped instead of boundary-marked, the output of the transducer would be [dæm]n. However, in [dæm][ne][šən], the first [n] would be parsed into a new onset, allowing it to be pronounced.

7 Polymorphemic Words

Some polymorphemic words will not be treated properly given the description above. For instance, **transplant** will receive the unorthodox syllabification [træn.splænt]. Assuming the system knows of morpheme boundaries, a preference for syllabifications where syllable breaks coincide with morpheme boundaries can be stated. This is accomplished by the upper-side priority union operator [13].

```
define Syllabify [
[sRule .o. NasRule .o. MainRule .o. SyllableWellFormedness]
                              .P.
[IgnoreMorphBoundaries .o. sRule .o. NasRule .o. MainRule]
];
```

We also define a SyllableWellFormedness filter that disallows parses where a syllable violates the the well-formedness of onsets or codas in English:

```
define SyllableWellFormedness [[SSP "."]* SSP];
define IgnoreMorphBoundaries "|" -> 0;
```

The motivation for the .P. construction is to allow words that would syllabify correctly when morpheme boundaries are treated as syllable boundaries. The syllabification [træns.plænt] contains no illegal onsets or codas, and is accepted. But there are words where morpheme boundaries cannot be respected without incurring an illegal onset, e.g. **deca**$|_{\mu}$**(a)thlon** should not yield [dɛ.kæ.θlɑn] since the sequence [θl] is not a well-formed onset in English. The first part of the rule in this case will have no output (it is blocked by SyllableWellFormedness) since [θl] is not among the legal onsets, and is prevented by well-formedness filter. The priority union operator ensures that only the lower rule cascade applies if the output language of the upper rule is 0, giving in this case the correct final output [dɛ.kæθ.lɑn]. The lower rule simply removes the morpheme boundary markers, and syllabification proceeds normally.

8 Implementing Alternative Approaches

The phonological literature is rife with differing proposals for the syllabification of English, and agreement seems to be rare. This is why we chose a standard

source whose syllabifications seemed natural (M-W), and the principles of the syllabifier were then developed according to this specific set of empirical data.

This results in a fairly conservative and traditional view of English syllabification—one that does not allow more complex phonological representations such as ambisyllabicity (where a single consonant is seen to belong to two adjacent syllables, as in Kahn's treatment [1]), or gemination (where a single consonant is represented as two segments, following e.g. Hammond [9]).

Most approaches to English syllabification are implementable with the basic methods outlined here. Four other approaches were encoded as FSTs to compare their respective predictions. These were the generative views of Kahn [1] and Selkirk [14], as well as the more recent Optimality Theory based views in Hammond [9], and Hall [11]. This simplicity of implementation crucially hinges on the existence of a shortest-replace operator (@>) and the upper-side priority union operator. Defining these through more primitive operators would severely complicate the task of constructing correct transducers.

When implemented as FST rewrite rules, the generative approaches were shown to be quite similar, differing only in the minutiae of the rewrite rules, despite the fact that the original descriptions often follow an involved formalism. However, these small differences often lead to wide variety of predictions, as seen in table 6.

Table 6. A sampling of the differing views on the English syllable. The second column represents the predictions made by the implementation described here. It should be noted that many of the examples here are not provided by the original authors—rather, a finite-state syllabifier has been reconstructed based on information given by the original sources. In the phonological literature, many details are often abstracted away from, and some essentials are presumed to be known, such as the set of allowed onsets. Often such details must be inferred from the specific examples given by the authors.

		Kahn (1976)	**Hammond (1999)**	**Hall (2004)**
feisty	fɑys.ti	[fɑy[s]ti]	fɑyst.i	fɑy.sti
cascade	kæs.ked	[kæ][sked]	kæs.sked	kæ.sked
pity	pɪ.ti	[pɪ[t]i]	pɪt.i	pɪ.ti
vanity	væ.nə.ti	[væ[n]ə][ti]	væn.ət.i	væ.nə.ti
texture	tɛks.čr̩	[tɛks][čr̩]	tɛks.čr̩	tɛk.sčr̩

9 Concluding Notes

We have presented general strategies to handle syllabification by finite-state means, as well as the details of an English syllabifier (see table 7 for examples of the output). The particular implementation is compact and the end result is a transducer with 52 states if the special handling that respects morpheme boundaries is ignored, and 188 states with this addition. This compares favorably with optimality theoretical implementations we have also evaluated as a

comparison—the smallest of which (following Hall [11]), using the construction method given by Gerdemann and Van Noord [15] is minimally represented by 1768 states.

Table 7. Example outputs of the syllabifier. No morpheme boundaries were present in the input.

acquiesce	æ.kwɪ.ɛs	**aspen**	æs.pɛn
atrocious	ə.troˌ.šəs	**atrophy**	æ.trə.fi
comptroller	kəmp.troˌ.lər	**computer**	kəm.pyu.tər
deluge	dɛl.yuǰ	**esquire**	ɛs.kwəyr
establishment	ɪs.tæ.blɪš.mɛnt	**exclaim**	ɛks.klem
explain	ɛk.splen	**exquisite**	ɛk.skwɪ.zət
extra	ɛk.strə	**formula**	for.myʊ.lə
gestation	ǰɛs.te.šn̩	**inkling**	ɪŋ.klɪŋ
manipulate	mə.nɪ.pyʊ.let	**manual**	mæn.yu.l̩
mattress	mæ.trəs	**metro**	mɛ.troˌ
Mississippi	mɪ.sə.sɪ.pi	**mistrust**	mɪs.trʌst
tenuous	tɛn.yu.əs	**transcribe**	træn.skrayb
venue	vɛn.yu	**Venusian**	vɛ.nu.šn̩

References

1. Kahn, D.: Syllable-based Generalizations in English Phonology. PhD thesis, MIT (1976)
2. Hooper, J.: Constraints on schwa-deletion in American English. In Fisiak, K., ed.: Recent Developments in Historical Phonology. Mouton, The Hague (1978) 183–207
3. Chomsky, N., Halle, M.: The Sound Pattern of English. Harper and Row (1968)
4. Kaplan, R.M., Kay, M.: Regular models of phonological rule systems. Computational Linguistics **20** (1994) 331–378
5. Beesley, K., Karttunen, L.: Finite-State Morphology. CSLI, Stanford (2003)
6. Laaksonen, K., Lieko, A.: Suomen kielen äänne- ja muoto-oppi [Finnish Phonology and Morphology]. Finn Lectura (1998)
7. Kenstowicz, M.: Phonology in Generative Grammar. Blackwell (1994)
8. Clements, G.N., Keyser, S.J.: CV Phonology: A Generative Theory of the Syllable. MIT Press (1983)
9. Hammond, M.: The Phonology of English. Oxford (1999)
10. Coleman, J., Pierrehumbert, J.: Stochastic phonological grammars and acceptability. Proceedings of the 3rd Meeting of the ACL Special Interest Group in Computational Phonology (1997) 49–56
11. Hall, T.A.: English syllabification as the interaction of markedness constraints. ZAS Papers in Linguistics **37** (2004) 1–36
12. Blevins, J.: The syllable in phonological theory. In Goldsmith, J.A., ed.: The Handbook of Phonological Theory. Blackwell (1995)
13. Karttunen, L.: The proper treatment of optimality theory in computational phonology. In: Finite-state Methods in Natural Language Processing, Ankara (1998) 1–12

14. Selkirk, E.O.: The syllable. In: Phonological Theory: The Essential Readings. Blackwell (1999)
15. Gerdemann, D., van Noord, G.: Approximation and exactness in finite state optimality theory. In Jason Eisner, Lauri Karttunen, A.T., ed.: Proceedings of the Fifth Workshop of the ACL Special Interest Group in Computational Phonology. (2000)