# Unsupervised Text Syllabification with Information from Audio

*Simple4All Internship Report, September 2013-February 2014*
*Technical University of Cluj-Napoca, Romania*

Popescu Dalia Georgiana
Supervisors: Mircea Giurgiu, Adriana Stan

*The objectives of this internship were: studies on supervised and unsupervised text syllabification techniques, with a focus on unsupervised methods, implementation and evaluation of several unsupervised audio syllabification algorithms, implementation and evaluation of an unsupervised text syllabification algorithms, improve the performance of text syllabification with features from acoustics.*
*In this report I shortly present my work as well as the results obtained and the conclusions I have on the topic.*

## I. Supervised versus unsupervised syllabification

As it has been showed in many studies and out team observed, syllabification improves the quality of speech synthesis as the rhythm is better defined and f0 seems more natural.

a) Syllabification using data driven approaches

Syllabifying purely based on the syllabification principles is sometimes inadequate as in practice the rules may be insufficient to disambiguate in some situations. More than that, there are cases when correct syllabification breaks one of the principles. Hence a more fine grained interpretation, typically some statistical formulation of these principles and the combination thereof, has given rise to a number of data driven syllabification methods that mainly differ in how these principles are incorporated in a model and how the model parameters are estimated from a corpus of example data.

Muller [1] developed grammars to describe the phonological structure of words. To increase the prediction precision of syllable boundaries, she introduces fine-grained grammars to better learn the phonotactic information. Using grammars, a word is presented as a syllable sequence. Each syllable splits into an onset and a rhyme. The rhyme at the same time is written as a nucleus and a coda. Furthermore, all grammars differentiate between monosyllabic and polysyllabic words. Additionally, the grammars distinguish between consonant clusters of different size.

Another approach was suggested by Zhang and Hamilton [2]. They presented the LE-SR (Learning English Syllabification Rules) system, which learns rules using a symbolic pattern recognition approach.

Each grapheme in a word is translated into C-S-CL representation. “C” stays for a consonant; “S” - for a syllabic grapheme and “CL” - for a consonant cluster. Syllabification rules and cutting patterns are learned though a syllabified corpus. To determine which cut should be chosen as a candidate rule, the authors combine a statistical approach with a symbolic pattern recognition approach and calculate the frequency of each cut.

Ananthakrishnan [3] looked at the syllabification problem as searching for the most probable syllable bracketing given a phoneme sequence. The author used a statistical approach with supervised and unsupervised learning. He simplified the probability of a syllabification given the nuclei to the product of probabilities seeing the onset and coda given the previous, current and following nucleus. This method bears some resemblance to ours; however, it employs a different model parameterization and parameter estimation approach.

Bartlett [4] suggested a discriminative approach that combines Support Vector Machine and Hidden Markov Model technologies and achieved one of the best published results.

A multiclass SVM was used to classify each phoneme according to its position in a syllable on the basis of a set of features. The HMM overcomes the problem of treating each phoneme in a word independently. When training a structured SVM, each training instance (word) is paired to its label (syllabification as sequence of onset/nucleus/coda), drawn from the set of possible labels. The SVM finds the best separator between correct and incorrect tagging.

b) Unsupervised language-independent syllabification

Several methods have been proposed in the literature for an unsupervised language-independent syllabification. Some methods that have been suggested in the literature rely on the observation that word-medial consonant clusters mostly constitute a subset of word-peripheral clusters. Intervocalic consonant clusters can therefore be divided up into a word-final (coda) and word-initial cluster (coda). Theoretically, two types of problems can be encountered: those where more than one division is possible and second, those in which no division is possible.

Several approaches have been suggested to resolve the first problem, i.e., word-medial consonant sequences where there are several possible divisions based on the occurrence of word-initial and word-final clusters. O’Connor and Trim [5] and Arnold [6] suggest that in cases of ambiguous word-medial clusters the preference for one syllable division over another can be determined by the frequency of occurrence of different types of word-initial and word-final clusters. For this purpose, they determine the frequency of occurrence of word-initial and word-final CV, VC, etc. syllable patterns. Based on these frequencies they calculate the probabilities of dividing a word-medial sequence by summing up the values established for the different word-peripheral syllable types. The candidate syllabification with the highest sum is then chosen as the optimal division.

In his article, [7] Thomas Mayer uses a slight modification of the proposal in O’Connor and Trim [5] and Arnold [6]. Instead of counting the frequency of occurrence of syllable types, the actual syllables are counted in order to determine the best split of word-medial consonant sequences.

As Thomas Mayer observed one additional problem when working with written texts rather than transcribed corpora is the act that diphthongs are not clearly distinguished from sequences of monophthongs. Yet this is vital for a correct syllabification procedure since the number of syllables of the word is different depending on this choice. In order to retrieve the diphthongs of the language from the distribution of vowel sequences in the corpus he used the following approach. For each bigram vowel sequence the number of times the first vowel v1 is directly followed by the second vowel v2 is compared with the number of times both vowels are separated by one consonant. If the frequency of direct adjacency is higher than the frequency of v1 cv2 the sequence is considered to be a ”diphthong”; if not, the sequence is considered to be a case of hiatus and both vowels are attributed to different syllables. Similar to Sukhotin’s algorithm the present syllabification algorithm is also global in the sense that the diphthong/monophthong distinction is always used in the same way no matter in which environment the sequence occurs.

Mayer tested his method on a manually created gold standard of 1,000 randomly selected words in Latin. The precision was 92.50% and the recall 94.96% (F-Score 0.94) for each transition from one symbol to another. Most misplaced syllable boundaries were due to the vowel cluster “io”, which has been treated as a diphthong by his method.

In their paper, Kseniya Rogova and Kris Demuynck [8], present a statistical approach for the automatic syllabification of phonetic word transcriptions. Their approach combines a probabilistic formulation of the legality, sonority and maximal onset principles with co-occurrence statistics in a single model.

## II. Implementation and evaluation of several unsupervised audio syllabification algorithms

### a)Praat script for syllable nuclei

De Jong and Wempe [2] used a Praat script to to detect syllable nuclei and measure speech rate automatically . They used intensity first to find peaks in the energy contour, since a vowel within a syllable (the syllable nucleus) has higher energy than the surrounding sounds. They then used the intensity contour to make sure that the intensity between the current peak and the preceding peak is sufficiently low. With this procedure, they deleted multiple peaks within one syllable. Finally, they used voicedness to exclude peaks that were unvoiced, which is required to delete surrounding voiceless consonants that have high intensity.

I implemented the algorithm and evaluated it on 100 audio files with manually created text grids. The accuracy of the algorithm was evaluated using two methods.

The first evaluation method computes the accuracy as:

Number of found syllables*100/Total number of syllables;

The second evaluation method computed the accuracy as:

Accuracy=(N-D-S-I)*100/N;

Where N=total number of manual markers, D is the number of marker deletions, I is the number of inserted markers, S is the number of automatic markers shifted from their corresponding manual markers.

The evaluation results computed with the above methods are:

| Method | Results |
|---|---|
| first | 81.59% |
| second | 59.77% |

Table 1. Praat method evaluation

As it can be seen in Table 1 the algorithm has good results when evaluated with a raw evaluation method and less good results when evaluated with a more critical method.

**b) Sonority function algorithm**

The algorithm for computing sonority is described in [3].

The algorithm for the detection of syllable boundaries using the sonority function is represented in the schema below:

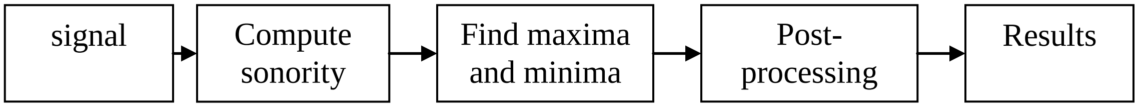

Figure 1. Sonority algorithm for detecting syllable boundaries

I implemented the above described algorithms and tested them on 10 audio files.

I evaluated the algorithm with the same methods used for evaluation of the praat nuclei script.

| Method | Results |
|---|---|
| first | 80% |
| second | 25% |

Table 2. Sonority syllabification evaluation

The results were good but the main sources of errors of the algorithm were the diphthong/hiatus groups, the groups of interest for my research.

### III. Implementation and evaluation of an unsupervised text syllabification algorithms

Both Mayer and Maximum Onset Principle Algorithms are based on extracting legal onsets from an unsyllabified corpus. After this procedure is performed only once for a certain language, the syllabification can be performed without any further data for any word, sentence or text. This is a big advantage of the two algorithms.

### a)Mayer syllabification

I implemented Mayer syllabification algorithm and performed several test on both English and Romanian. The results obtained are presented below:

| Language | Test data | Accuracy percentage |
|---|---|---|
| English | 7880 words from Celex | 24.25% |
| English | 108674 words from dictionary | 27.8% |

Table 3. Mayer algorithm evaluation

As it is presented in Table 3, the results are not satisfactory, mainly because of the phonetic transcription of English.

### b)Maximum Onset Principle(MOP) syllabification

Maximum Onset Principle states that all vowels are (in most cases) syllable nuclei. So, the problem is with the intervocalic consonants. The algorithm takes each group of consonants between two vowels and tries to maximize the second's syllable onset. The coda can be very small or not to exist at all.

The algorithm also respects the legality principle. The onset can be maximized only if it is a legal onset.

Legal onsets are prior determined as being all the consonants group that precede the first vowel in a word.

Diphthong/hiatus problem is resolved in the same manner as in Mayer's algorithm, by calculating each two vowels frequency as v1v2 or v1cv2.

The results obtained with MOP syllabification are presented in Table 4.

| Language | Test data | Accuracy percentage |
|---|---|---|
| English | 7880 words from Celex | 30.12% |
| English | 108674 words from dictionary | 37.1% |
| Romanian | 500000 word from dictionary | 50% |
| Romanian | 500 sentences of common speech | 80.48% |
| Finnish | 5000 words | 73% |

Table 4. MOP algorithm evaluation

As it presented numerical in Table 3 and Table 4, MOP algorithm showed better performances than Mayer’s algorithm in English and very good performances with Romanian and Finnish. The decision was to continue the work with MOP algorithm.

A suggestion for improving the syllabification technique for English is to use Maximum Onset Principle on the acoustic transcription of the words. Tests were not possible due to the lack of test data.

After studying the source of errors, the conclusions were that, for Romanian, more than 50 percent of the errors (60% for the dictionary, 55% for rnd1) were produced by the diphthong/hiatus confusion.

## IV. Improve the performance of text syllabification with features from acoustics

As the diphthong/hiatus problem could not be resolved with the information from the text, the next thing is to use information from acoustics to perform the correct syllabification.

### a)MOP plus audio information from Praat nuclei script

The first approach was to use audio syllabification information to solve the diphthong/hiatus problem generated from MOP text syllabification.

I evaluated the algorithm on rnd1 from RSS. The syllabification accuracy without Praat information was 80.42% and with the Praat information was 80.8%. The improvement was only 0.4% so the conclusion was that the algorithm was not accurate enough.

### b)Diphthong/hiatus features

Researcher have tried to prove [14] that diphthong groups and hiatus groups have specific spectral and audio features that could help identify them in a audio/spectral analysis. They expect for hiatus to have a longer duration that the diphthong, a greater segmentation in the spectrum, reflected in the MFCC coefficients, and a dip in F0.

I elaborated an algorithm for diphthong/hiatus extraction from rnd1 audio files.

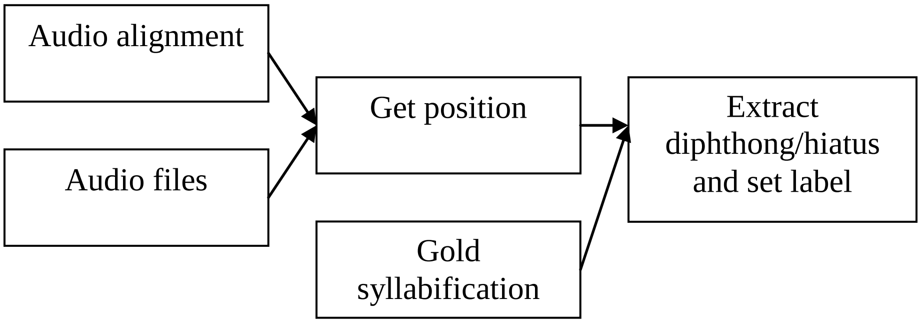

Figure 2. Extraction algorithm

With the audio segments extracted and labeled we identified relevant features for diphthong/hiatus group. These are:

- 13 deltaMFCC coefficients ;
- 4 deltaFormants;
- Delta F0
- Delta duration (expected for the hiatus to have a larger duration)

**c)Diphthong/hiatus classification with Weka**

Using the 19 extracted features mentioned above, we used Weka in order to classify the diphthong/hiatus groups. Weka is a collection of machine learning algorithms for data mining tasks.

We used decision trees algorithm CART J.48 and the results obtained with 10 fold cross validation are presented in the table below.

| Test data | Results |
|---|---|
| Rnd1 | 92% |
| Rnd2 | 73% |
| Rnd3 | 76% |
| Rnd1+Rnd2+Rnd3 | 80% |

Table 5. Weka classification results

In Table 5, the presented data shows that the results obtained were good and very good, meaning that this is a promising approach in resolving diphthong/hiatus problem.

Results obtained with KMeans, Naïve Bayesian Classifier or other classifiers provided by Weka were poor (hardly over 50%).

**Conclusions and further work**

The importance of syllabification in generating natural synthesized speech is proven to be great.

Syllabification has been analyzed and implemented by many researchers along the years but, until this day, a totally unsupervised approach with good results has not been found.

In my activity I studied the unsupervised text syllabification performed with Maximum Onset Principle and obtained very promising results for languages where the phonetic transcription is close to the text.

I observed through the test performed that the main source of errors for this algorithm is the diphthong/hiatus problem.

Having extracted the audio groups of diphthong and hiatus, I used Weka decision trees and performed the classification based on 19 features. The results obtained have a great accuracy and indicate that this is a good approach in resolving diphthong/hiatus problem.

**The work can be extended by:**

- Identifying other parameters relevant for the classification
- Identifying methods for classification that do not use training data
- Supplying the training data via user-feedback
- Identifying and solving other sources of errors of MOP syllabification algorithm.

**Bibliography**

**[1] Muller, Karin (2006),** "Improving syllabification models with phonotactic knowledge", *Proceedings of the Eighth Meeting of the ACL Special Interest Group on Computational Phonology at HLT-NAACL, pp. 11–20.*

**[2] Zhang, Jian and Howard J. Hamilton** (1997), "Learning English syllabification for words", *ISMIS, pp. 177–186.*

**[3] Ananthakrishnan, Shankar** (2004), "Statistical syllabification of English phoneme sequences using supervised and unsupervised algorithms", *Technical report, CS562 Term Project Report.*

**[4] Bartlett, Susan, Grzegorz Kondrak, and Colin Cherry** *(2009), "*On the syllabification of phonemes", *HLT-NAACL, The Association for Computational Linguistics, pp. 308–316.*

**[5] J. D. O'Connor and J. L. M. Trim.** *1953. "*Vowel, consonant, and syllable - a phonological definition". *Word,9(2):103–122.*

**[6] Gordon F. Arnold.** *1955-1956. "*A phonological approach to vowel, consonant and syllable in modern french*." Lingua, V:251–287.*

**[7] Nivja H. de Jong**, Ton Wempe "Praat script to detect syllable nuclei and measure speech rate automatically*", Behavior Research Methods, May 2009, Volume 41, Issue 2, pp 385-390*

**[8] Thomas Mayer**, "Toward a Totally Unsupervised, Language-Independent Method for the Syllabification of Written Texts" *, Proceeding of SIGMORPHON '10 Proceedings of the 11th Meeting of the ACL Special Interest Group on Computational Morphology and Phonology*

**[9] Bogdan Ludusan**, "Beyond short units in speech recognition: a syllable-centreic and prominence-centric approach " *PhD Thesis, Universita degli studi di Napoli "Federico II"*

**[9] Liviu P. Dinu**, "Departirea Automata in Silabe a Cuvintelor din Limba Roamana. Aplicatii in Constructia Bazeide Date a Silabelor Limbii Romane" *Raport de Cercetare, University of Bucharest, Faculty of Mathematics*

**[10] Adriana Cornelia STAN**, "Romanian Hmm-Based Text-To-Speech Synthesis
With Interactive Intonation Optimisation" *PhD Thesis, Technical University of Cluj-Napoca, faculty of Electronics, Telecommunications and Information Technology*

**[11] David Rybach,** Christian Gollan, Ralf Schl¨ter, Hermann Ney, "Audio Segmentation for Speech Recognition Using Segment Features*", Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on, 19-24 April 2009, 4197 - 4200*

**[12]** ***http://ilr.ro/silabisitor/***

**[13]** ***http://www.cs.waikato.ac.nz/ml/weka/index.html***

**[14] Ayaz Keerio, Lachhman Das Dhomeja, Asad Ali Shaikh, Yasir Arfat Malkani, '**Comparative Analysis of Vowels, Diphthongs and Glides of Sindhi'*, Signal & Image Processing : An International Journal (SIPIJ) Vol.2, No.4, December 2011*

*The activity leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement No 287678 (www.simple4all.org) in the "Simple4All Internship Programme". The activities have been carried out using data and computing facilities available at the "Speech Processing Group (Http://speech.utcluj.ro)" from The Technical University of Cluj-Napoca.*