Syllabification Rules Versus Data-driven Methods in a Language with Low Syllabic Complexity:
The Case of Italian

Connie R. Adsett, Yannick Marchand, Vlado Kešelj

Abstract

Linguistic rules have been assumed to be the best technique for determining the syllabification of unknown words. This has recently been challenged for the English language where data-driven algorithms have been shown to outperform rule-based methods. It may be possible, however, that data-driven methods are only better for languages with complex syllable structures. In this study, three rule-based automatic syllabification systems and two data-driven automatic syllabification systems (Syllabification by Analogy and the Look-Up Procedure) are compared on a language with lower syllabic complexity — Italian. Comparing the performance using a lexicon containing 44,720 words, the best data-driven algorithm (Syllabification by Analogy) achieved 97.70% word accuracy while the best rule set correctly syllabified 89.77% words. These results show that data-driven methods can also outperform rule-based methods on Italian syllabification, a language of low syllabic complexity.

Key words: syllabification, Italian language, rule-based systems, data-driven methods, analogy

1. Introduction

It is widely held that syllables are an important aspect of language. Not only are they used as a necessary unit in phonetics (Laver, 1994) and in phonological theory (Fudge, 1969; Hooper, 1972; Selkirk, 1982), they are also thought to play a key role in the processing of spoken language. This is substantiated by the results of both Tabossi et al. (2000) and Finocchiara and Bertinetto (2003) which suggest that speakers of Italian rely on the syllabic structure when accessing the

*Corresponding author

Email addresses: adsett@cs.dal.ca (Connie R. Adsett), yannick.marchand@mrc-cnrc.gc.ca (Yannick Marchand), vlado@cs.dal.ca (Vlado Kešelj)
mental lexicon to process spoken Italian. For English speakers, the findings of Bruck et al. (1995) indicate that syllabified representations of spoken nonwords are constructed in phonological memory in order to perform comparisons with additional nonwords, even though English is believed to have more syllabic irregularity.

The concept of varying levels of syllabic regularity amongst languages is a generally well-understood concept. It has been used to support results such as the similarities and differences in visual word recognition studies between Spanish and English (Álvarez et al., 2001; Carreiras et al., 1993) and Spanish and German (Conrad and Jacobs, 2004), the lack of a rigid internal syllable hierarchy in Italian as compared to English (Bertinetto et al., 1994) and the segmentation strategies used for processing spoken words in Catalan and Spanish (Sebastián-Gallés et al., 1992) and French and English (Bruck et al., 1995; Cutler et al., 1986).

There is no well established means of comparing the syllabic complexity of languages. However, methods which have been used can be conceptually divided into two categories: structural and behavioral. The structural approach is adopted by Dauer (1983) as one of the components in the method of classifying languages as stress-timed or syllable-timed. In this method, the variety and frequency of permissible syllable types are used as the means of placing languages on the scale between stress and syllable-timed. Dauer (1983) found that languages which tended toward the syllable-timed end of the spectrum (those also held to have simpler syllabic complexity) had a majority of CV syllables (where each C denotes a single consonant and each V represents a single vowel), and had more open syllables than closed. Stress-timed languages, such as English, have, by comparison, fewer CV syllables and a wider variety of both open and closed syllables. The statistics reported by others for six European languages allow for the comparison of CV structure frequency shown in Figure 1.

The behavioral method of comparing the syllabic complexity of languages endeavors to distinguish between syllabically simple and complex languages through human performance. Using this approach, Seymour et al. (2003) found a significant difference in the ability to read non-words for children in grades one and two. This difference corresponded to the distinction between simple and complex syllable languages. The languages with simple syllabic structure (Finnish, French, Greek, Italian, Spanish, and Portuguese) had significantly lower error rates on non-word reading than those languages with complex syllabic structure (Austrian, Danish, Dutch, English, German, Icelandic, Norwegian, and Swedish).

Regardless of the syllabic structure of a language, rules are generally believed to be the appropriate method to be used for the syllabification of words (Cioni, 1997; Dumay et al., 2002; Floccia et al., 1999; Goslin and Floccia, 2007; Laganaro and Alario, 2006; Levelt et al., 1999). Such rules take a similar form to that of the phonological rule structure described by Damper et al. (1999). Given a left-context $A$ and a right-context $B$, a juncture (a position between two adjacent characters in a word) may be a syllable boundary or not. For example, $AB \rightarrow A - B$ would be a rule which places a syllable boundary between $A$ and
Figure 1: Frequency of CV syllables in Dutch (Frota and Vigário, 2001), English (Dauer, 1983), European Portuguese (EP) (Frota and Vigário, 2001), French (Laks, 1995), Italian (Bortollini, 1976), and Spanish (Dauer, 1983).

These left and right contexts are typically fewer than three characters in length (see Sections 3.1-3.3 for some examples). They are generated by experts in the language in question and form a fixed set, once derived. Notions such as sonority theory (Clements, 1990) and the maximal onset principle (Kahn, 1976; Pulgram, 1970) have typically been used as the foundations for the descriptions of such rule sets.

Although various aspects of these principles have been tested against human syllabification, it is seldom that these rules have been tested and validated against large lexicons of syllabified words. This situation is surprising given that rules that have been explicitly described can be easily implemented and tested by a computer. The work of Cuayáhuítl (2004) is, nevertheless, a notable counterexample which evaluates a set of Spanish rules on a lexicon of 316 words and reports 98.4% word accuracy. Reichel and Schiel (2005) also evaluate a set of syllabification rules for German as part of their work on grapheme-to-phoneme conversion. They report 81.4% word accuracy when no morphological information is used and 92.48% word accuracy with morphological information.

In a recent comparison of English syllabification rules and data-driven methods of automatically determining syllabification, the data-driven methods outperformed the rule sets by a significant amount (Marchand et al., in press). These computational results support the view put forth by Kessler and Treiman (1997, page 295) that “abandoning the idea that only absolute, inviolable restrictions are worthy of note” may be necessary. Their further statement that the “recognition and production of a word can be affected by how many other
words are similar to it in pronunciation or spelling” (Kessler and Treiman, 1997, page 297) supports the possible advantages of data-driven algorithms which use known examples stored in the internal lexicon to determine the syllabification of unknown words.

However, both the works of Marchand et al. (in press) and Kessler and Treiman (1997) are based on English. In fact, the reason for the claims made by Kessler and Treiman (1997) is that English has a greater syllabic complexity than other languages. Even in their work, it is assumed that the patterns found in English are established as absolute rules in languages with simpler syllabic structure. The success of data-driven methods on this language may be due to the fact that English is a language with a complex and irregular syllable structure (Bruck et al., 1995; Conrad and Jacobs, 2004), which is challenging (and perhaps impossible) to fully capture with traditional linguistic rules. In order to test whether this is true, it is also necessary to compare rule sets to data-driven syllabification algorithms for a language with lower syllabic complexity.

The low syllabic complexity language used in this comparison was required to have a freely available and reliable syllabified lexicon along with published rule-based methods. The Italian language met this criteria. Not only does it have low syllabic complexity (Bertinetto et al., 1994; Ramus et al., 1999; Seymour et al., 2003), a large syllabified lexicon was available (Cosi et al., 2001), along with three rule-based methods previously used in research on the Italian language (Cioni, 1997; Hall, 1974; MacKinney-Romero and Goddard, 2006). As stated above, the CV syllable has been found to occur with a frequency of 60% in Italian, the highest frequency of those languages reported in Figure 1 and Italian was among those languages for which Seymour et al. (2003) found childrens’ nonword reading to be more accurate and rapid. In this study, we therefore chose to compare Italian syllabification rules to the best two data-driven methods used in the previous comparison on the English language (Marchand et al., in press).

After describing the Italian lexicon used in Section 2 and presenting the rule-based and data-driven methods tested (Sections 3 and 4, respectively), the experiments performed are described in three studies (Sections 5 to 7). In Study 1 we compared the performance of the syllabification rules with that of the data-driven algorithms. An examination of the impact of lexicon size on the performance of the data-driven methods was performed for Study 2. Finally, in Study 3 the results on Italian syllabification were compared with the findings for English.

2. Italian lexicon

The Italian lexicon used in this article is part of the Italian Festival TTS project (Cosi et al., 2001). It was created by Instituto Trentino di Cultura — Istituto per la Ricerca Scientifica e Tecnologio (ITC-irst) and Instituto di Scienza e Tecnologi della Cognizione — Sezione di Padova “Fonetica e Dialettologia” —
Each entry in this lexicon provides spelling, part-of-speech, pronunciation, stress, and syllabification information. This is the format used in the Festival Speech Synthesis System (Clark et al., 2004), for which this lexicon was created. In the following example:

("sempre" B (((s E1 m) 1) ((p r e) 0)))

- “sempre” (Italian for ‘always’) is the word;
- B is the part-of-speech label (denoting an adverb);
- “sE1mpre” is the sequence of phonemes which represent the pronunciation of the word;
- “sE1m” is the first syllable;
- the 1 following “(s E1 m)” indicates that this syllable receives the primary stress;
- and “pre” is the second syllable (which is not stressed, as shown by the digit 0).

A total of 440,084 entries exist in the original lexicon. Because there is such similarity in Italian between different forms of each lexeme, we endeavored to reduce the lexicon to only one form of each lexeme. For this reason, plurals, verb forms (apart from the infinitive), superlatives, and comparatives were removed from the lexicon, along with homophones and homographs. Finally, to be consistent with previous work, proper nouns were also removed, leaving 53,399 words.

Because syllabification information was given in the pronunciation domain and the three Italian rule-based syllabification algorithms operate on the spelling domain (see Section 3), all words which did not have the same number of letters as phonemes were also removed. This allowed syllable boundaries to be easily transferred to the spelling domain without any need for a complex alignment processing (for example, “sE1m—pre” becomes “sem—pre”). Using this simple alignment approach, only 8,697 words were removed. The resulting lexicon (referred to as the **Full Italian lexicon** below) consisted of 44,720 entries. It is unknown whether words with different numbers of letters and phonemes differ in syllabic complexity and how this could affect the results of the automatic syllabification algorithms.

The distribution of words according to their part of speech (for the four most common types) is given for both the original lexicon and the Full Italian lexicon (Figure 2). The differences are as expected. The proportion of verbs decreases

---

1. This lexicon is freely available at [www.pd.istc.cnr.it/Software/It-Festival/2.0/lex_ifd.zip](http://www.pd.istc.cnr.it/Software/It-Festival/2.0/lex_ifd.zip) (last accessed 22 July 2008).  
2. The SAMPA phonetic alphabet is used in this lexicon with the addition of the symbol ‘1’ to indicate stressed vowels (Cosi et al., 2001).
substantially because this is the area in which the majority of the similar words existed (the same verb with different conjugations). The Full Italian lexicon also contains a wide range of word lengths (shown in Figure 3) — from those consisting of only one character to one with 26 (precipitevolissimevolmente; *very hurriedly*).

![Figure 2: Distribution of parts of speech in the original and Full Italian lexicons.](image)

The frequency of the most common syllable structure types found in this lexicon is shown in Figure 4. According to this data, the Italian lexicon used provides a representation consistent with the distribution found by Bortollini (1976) in his statistical analysis of syllable structures found in the Italian language (t-value = 0.016, p > 0.95).

### 3. Rule-based algorithms

Three Italian-specific rule-based algorithms for automatic syllabification were tested. The algorithms selected were: Cioni’s algorithm for the syllabification of written Italian (Cioni, 1997), an implementation of Hall’s ordered rules for Italian syllabification (Hall, 1974), and Bergamini’s SYL-LABE syllabification algorithm (MacKinney-Romero and Goddard, 2006). Although formally defined, none of the methods have been tested or compared in order to determine the accuracy of their rules. All three algorithms operate on the spelling domain.
3.1. Cioni’s algorithm for the syllabification of written Italian

Cioni (1997) presents a method using what he claims to be a “minimal set of rules” developed with the assistance of Italian linguists.³

In his description of the implementation of this method, a subset of the rules used by Cioni (1997) are listed below:

1. CVCV → CV | CV;
2. VC1C2V → VC1 | C2V, if C1 = C2;
3. VC1C2V → V | C1C2V, if C2 = h;
4. VCCV → VC | CV;
5. VC1C2C3V → VC1 | C2C3V, if C1 ≠ s;
6. V1V2 → V1 | V2, if V1 ∈ {a,e,o} and V2 ∈ {a,e,o};
7. V1V2V3 → V1 | V2V3 if V1 ∈ {a,e,o};

³The source code (written in C) for this program is available from www.di.unipi.it/~lcioni/AltroSoftware/sillabatore.tar.gz (last accessed 22 July 2008).
Figure 4: Frequencies of Italian syllable structures found by (Bortollini, 1976) and in the Full Italian lexicon. Here ‘C’ represents a consonant and ‘V’ represents a vowel. The plus sign following the V indicates that more than one vowel may be present. Note that the values for the final syllable type (CCCV+C) are non-zero frequencies of 0.20% for the Italian Lexicon and 0.04% for Bortollini’s statistics.

8. $V_1V_2V_3 \rightarrow V_1|V_2|V_3$, if $V_1 = i$ and $V_2 \neq u$, or $V_1 = u$ and $V_2 \neq i$.

Rules are also included to specify which pairs of vowels form diphthongs and therefore cannot be separated into different syllables. All rules are applied recursively by searching through the given word from left to right.

3.2. Hall’s ordered rules for Italian syllabification

Hall (1974) lists six ordered rules for breaking single Italian words into syllables:

1. $C_1C_2 \rightarrow C_1|C_2$, if $C_1 = C_2$;
2. $C_1C_2 \rightarrow C_1|C_2$, if $C_1 = c$ and $C_2 = q$;
3. $C_1C_2 \rightarrow C_1|C_2$, if $C_1 \in \{m,n,r,l\}$;
4. $VCC \rightarrow V|CC$;
5. $VCV \rightarrow V|CV$;
6. never divide a sequence of vowels into multiple syllables.
He also provides two additional rules for division across word boundaries. The first states that a syllable boundary should never be placed immediately following an apostrophe which connects two words; for example, “l’albero” (‘the tree’). The second concerns placement of syllables in musical scores. In this context, when a final syllable in a word ends in a vowel and the next word begins with a vowel and they must be both sung on the same note or over tied notes, it is necessary to indicate that they form a single syllable. This is done by decreasing the space between the two syllables.

These rules are given with the intent of assisting Italian instructors in teaching students how to divide Italian words in the spelling domain. Because they are fully described, it was possible to implement a rule-based automatic syllabification program\(^4\) using these rules for the purpose of evaluation.

### 3.3. Bergamini’s SYL-LABE program

Bergamini’s rule-based syllabification algorithm is called SYL-LABE and was implemented in C\(^5\). The results of this algorithm were used as a gold standard in work on the automatic syllabification of Italian (MacKinney-Romero and Goddard, 2006). Two versions (1.0 and 3.3) of the algorithm are available and both were tested but only the results of the best algorithm (version 1.0) are reported. Implementation details are given in an Italian file\(^6\) which accompanies the download of version 3.3 of the program.

These rules place vowels in one of two categories, specified by Bergamini: strong (a, e, o) and weak (i, u). This gives rise to the rules:

1. \(V_1V_2 \rightarrow V_1V_2\), if \(V_1 \in \{a,e,o\}\) and \(V_2 \in \{i,u\}\);
2. \(V_1V_2 \rightarrow V_1V_2\), if \(V_1 \in \{i,u\}\) and \(V_2 \in \{a,e,o\}\);
3. \(V_1V_2V_3 \rightarrow V_1V_2V_3\), if \(V_1V_2V_3 = \{iai,uai,iei,uei,iuo,uoi\}\);
4. \(V_1V_2 \rightarrow V_1[V_2]\), in all other cases.

Additional rules used include:

1. \(VCV \rightarrow V(CV)\);
2. \(VC_1C_2 \rightarrow VC_1C_2\), if \(C_1 \in \{m,n,r,l\}\);
3. \(C_1C_2 \rightarrow C_1C_2\), if \(C_1 = C_2\);
4. \(C_1C_2 \rightarrow C_1C_2\), if \(C_1 = c\) and \(C_2 = q\);
5. \(VC_1C_2 \rightarrow V(C_1C_2)\), in all other cases.

The SYL-LABE program, as it was originally built, syllabifies only one word each time it runs. In addition to the word itself, stress information is required by the SYL-LABE algorithm in order to produce syllabified output. The results

\(^4\)A Python implementation of Hall’s rules is available from the authors upon request.

\(^5\)Bergamini’s program is available at [http://www.pierotofo.it/pages/sorgenti/C/Utility/](http://www.pierotofo.it/pages/sorgenti/C/Utility/) — last accessed 22 July 2008. Note that the program may not be listed on the first page of the site.

\(^6\)This file is called ‘Regole di silabazione.pdf’ and is included in the main folder of the zip file which constitutes version 3.3 of the program.
presented below were obtained using the stress information provided in the Italian lexicon. For example, the lexicon places the stress in the word “sempre” (‘always’) on the first syllable of the word (‘sem’). The position of the vowel in the stressed syllable is 2 because ‘e’ is the second character in the word. To run the SYL-LABE program, this information (sempre and 2) is given as input. A simple loop program was written in order to obtain the syllabification of a list of words using the SYL-LABE program.

4. Data-driven algorithms

The data-driven algorithms used in this comparison were the same two that performed best on the syllabification of English words (Marchand et al., in press): Syllabification by Analogy (SbA) and the Look-Up Procedure (LUP).

4.1. Syllabification by Analogy

Syllabification by Analogy is adapted from Pronunciation by Analogy, a method for automatic grapheme-to-phoneme transcription (Damper and Eastmond, 1997; Damper et al., 1999; Dedina and Nusbaum, 1991; Marchand and Damper, 2000; Yvon, 1996).

This method’s approach to the automatic syllabification of an unknown word is as follows. The word is first broken into all possible contiguous substrings of each possible length. These substrings are then compared to substrings of syllabified words in the lexicon to determine each segment’s syllabic structure. Possible syllabic structures for the substrings from the unknown word are obtained by finding the syllabic structures of matching substrings from words in the dictionary. When a match is found, the syllabification information for this substring is stored in a syllabification lattice (see Figure 5). The information stored in the lattice is then combined to construct the syllabification of the entire word.

The following example uses “sempre” as the unknown word and “semplice” (‘simple’) as the word from the lexicon. First, “sempre” is represented as the input string s?e?m?p?r?e where ? represents each position between letters (juncture) at which a syllable boundary may occur. Then the syllabified word “sem|pli|ce” is represented as s*e*m|p*l|i|c*e where * and | represent non-syllable and syllable boundaries, respectively. All possible contiguous substrings from the input string, s?e?m?p?r?e, (e.g., s?e, e?m, m?p, s?e?m, etc.) are then compared to the string s*e*m|p*l|i|c*e to find all matching substrings. Note that, when characters from the input string are compared to those from words in the lexicon, ? may be matched with either 1 or *.

The resulting syllabification lattice is a graph for which information from matching substrings form the nodes and arcs. Nodes represent the beginning and ending substring characters. Arcs are labeled with any intermediate substring characters along with the number of occurrences of this substring within the matches found in the dictionary. In the case of the current example, the nodes and arcs inserted into the lattice from the substring s*e*m|p would be $*$.
•_p, along with all other subsets of this (for example, •_s  \xrightarrow{s=1} •_e for s*e and •_s \xrightarrow{\pi=1} •_p for \pi|m|p).

All possible shortest paths through the lattice from the first to the last character of the input word are then identified. Figure 5 shows the resulting shortest paths for the word “tosasiepi” (Italian for ‘hedge shears’). Here, the shortest path is defined solely on structural characteristics. The weights of the graph edges are not considered at this stage, only the number of edges from the start node to the end node. Syllabification is obtained from a given path by concatenating the node and arc labels (aside from the frequencies). If only one shortest path is found, it is used to infer the syllabification of the unknown word.

Figure 5: Simplified example of a syllabification lattice using the word “tosasiepi” (Italian for ‘hedge shears’). For clarity, only the shortest paths are shown. The first arc indicates that the beginning of the word, namely T?O?S?A? has been matched seven times in the lexicon with the corresponding syllabification T*O|S*A:7.

When there is more than one shortest path, a set of scoring strategies is used to determine the best syllabification. The three scoring strategies that gave the highest performance on the English language (Marchand and Damper, 2007) were:

1. the product of the arc frequencies along the shortest path;
2. the frequency of the same syllabification within the shortest paths;
3. the weak link value where ‘weak link’ is the minimum of the arc frequencies.

For the sake of consistency, these same scoring strategies were used to determine the syllabification of Italian words. Table 1 shows how these scoring strategies are applied to the results for “tosasiepi”. After combining the scores of each strategy, the syllabification path with the best score is •_start \xrightarrow{t|o|s|a|e=7} •_s|a|e:42 •_end which gives the syllabification, “to|s|a|e|p”. Note that we omit any detailed description of how these strategies are combined since this fusion scheme has been extensively described in previous publications (Damper and Marchand, 2006; Marchand and Damper, 2000).
Table 1: Application of SbA scoring strategies to the shortest paths in the syllabification lattice shown in Figure 5.

<table>
<thead>
<tr>
<th>Strategy</th>
<th>Path</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1</strong></td>
<td><strong>t\rightarrow</strong> s\rightarrow i\rightarrow e\rightarrow p\rightarrow**</td>
<td><strong>1:1157</strong></td>
</tr>
<tr>
<td></td>
<td><strong>s\rightarrow i\rightarrow e\rightarrow p\rightarrow</strong></td>
<td><strong>8099</strong></td>
</tr>
<tr>
<td><strong>Product</strong></td>
<td><strong>t\rightarrow</strong> s\rightarrow i\rightarrow e\rightarrow p\rightarrow**</td>
<td><strong>1764</strong></td>
</tr>
<tr>
<td></td>
<td><strong>s\rightarrow i\rightarrow e\rightarrow p\rightarrow</strong></td>
<td><strong>84</strong></td>
</tr>
<tr>
<td><strong>of arc</strong></td>
<td><strong>t\rightarrow</strong> s\rightarrow i\rightarrow e\rightarrow p\rightarrow**</td>
<td><strong>49</strong></td>
</tr>
<tr>
<td></td>
<td><strong>s\rightarrow i\rightarrow e\rightarrow p\rightarrow</strong></td>
<td><strong>7</strong></td>
</tr>
<tr>
<td><strong>frequencies</strong></td>
<td><strong>t\rightarrow</strong> s\rightarrow i\rightarrow e\rightarrow p\rightarrow**</td>
<td><strong>3</strong></td>
</tr>
<tr>
<td></td>
<td><strong>s\rightarrow i\rightarrow e\rightarrow p\rightarrow</strong></td>
<td><strong>3</strong></td>
</tr>
<tr>
<td><strong>of same syllabification</strong></td>
<td><strong>t\rightarrow</strong> s\rightarrow i\rightarrow e\rightarrow p\rightarrow**</td>
<td><strong>3</strong></td>
</tr>
<tr>
<td></td>
<td><strong>s\rightarrow i\rightarrow e\rightarrow p\rightarrow</strong></td>
<td><strong>1</strong></td>
</tr>
<tr>
<td><strong>Weak link</strong></td>
<td><strong>t\rightarrow</strong> s\rightarrow i\rightarrow e\rightarrow p\rightarrow**</td>
<td><strong>1</strong></td>
</tr>
<tr>
<td></td>
<td><strong>s\rightarrow i\rightarrow e\rightarrow p\rightarrow</strong></td>
<td><strong>6</strong></td>
</tr>
<tr>
<td></td>
<td><strong>t\rightarrow</strong> s\rightarrow i\rightarrow e\rightarrow p\rightarrow**</td>
<td><strong>2</strong></td>
</tr>
<tr>
<td></td>
<td><strong>s\rightarrow i\rightarrow e\rightarrow p\rightarrow</strong></td>
<td><strong>1</strong></td>
</tr>
</tbody>
</table>
Table 2: Table entries used by the Look-Up Procedure for the word “s*e*m|p*l*i|c*e”

<table>
<thead>
<tr>
<th>N-gram</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>---semp</td>
<td>0 1</td>
</tr>
<tr>
<td>--semp1</td>
<td>0 1</td>
</tr>
<tr>
<td>-sempli</td>
<td>1 0</td>
</tr>
<tr>
<td>semplic</td>
<td>0 1</td>
</tr>
<tr>
<td>emplice</td>
<td>0 1</td>
</tr>
<tr>
<td>mplice-</td>
<td>1 0</td>
</tr>
<tr>
<td>plice--</td>
<td>0 1</td>
</tr>
</tbody>
</table>

4.2. Look-Up Procedure

The Look-Up Procedure was also originally used for grapheme-to-phoneme transcription (Weijters, 1991). It has since been modified to perform automatic syllabification (Daelemans and van den Bosch, 1992; Marchand et al., in press) and is classified by Daelemans and van den Bosch (2005) as a Memory-Based Learning approach. This method uses N-grams that correspond to N consecutive characters (each consisting of a left context, right context and central letter) to learn and determine syllable boundaries.

During training, an N-gram is generated for each possible syllable boundary location in a word. Each N-gram is stored in a table along with how often a syllable boundary occurs and does not occur following the central letter. Table 2 shows the table entries for the word s*e*m|p*l*i|c*e, using a left and right context of three letters (N = 7).

During testing, the closest matches to the N-grams from the test words are found in the table. Similarity between N-grams is determined using an N-element weight vector. For a given N-gram, if the frequency of a syllable boundary occurring after the central letter is higher than the frequency of no syllable boundary, a syllable boundary is placed in the test word.

For example, using the 7-grams stored in Table 2 to syllabify the word “sempre” requires finding the closest match to each of five 7-grams (---semp, --semp1, -sempli, semplic, emplice) within the table. Using [1, 4, 16, 64, 16, 4, 1] as the weight vector, the closest match to the third 7-gram given above (-sempre) is the entry -sempli with a similarity value of 101 (1+4+16+64+16). The frequency of a syllable boundary occurring after the central letter in this pattern is greater than the frequency of no syllable boundary and therefore a syllable boundary is placed following the m in “sempre”.

The Look-Up Procedure was tested with all 15 weight vectors (given in Table 3) that were used in the comparison of automatic syllabification methods for English (Marchand et al., in press) and the study in which this technique was originally described (Weijters, 1991).
Table 3: Weight vectors used in the Look-Up Procedure as used by (Weijters, 1991) and (Marchand et al., in press)

<table>
<thead>
<tr>
<th>Version</th>
<th>Left Context</th>
<th>Central Letter</th>
<th>Right Context</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-4</td>
<td>-3</td>
<td>-2</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>4</td>
<td>16</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>4</td>
<td>16</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>4</td>
<td>16</td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>4</td>
<td>16</td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td>4</td>
<td>16</td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>4</td>
<td>16</td>
</tr>
<tr>
<td>11</td>
<td>1</td>
<td>4</td>
<td>16</td>
</tr>
<tr>
<td>12</td>
<td>4</td>
<td>16</td>
<td>64</td>
</tr>
<tr>
<td>13</td>
<td>4</td>
<td>16</td>
<td>64</td>
</tr>
<tr>
<td>14</td>
<td>16</td>
<td>64</td>
<td>256</td>
</tr>
<tr>
<td>15</td>
<td>16</td>
<td>64</td>
<td>256</td>
</tr>
</tbody>
</table>

5. Study 1 — Data-driven methods versus rule-based systems

This study addressed the question of whether data-driven algorithms could be used to obtain better Italian syllabification accuracy than rules. Alternate gold standards, derived from the three sets of rules, were tested in addition to the Full Lexicon in order to ensure that the evaluation was not biased toward one lexicon’s syllable boundary placement.

5.1. Method

To compare the syllabification algorithms described above, a leave-one-out procedure was used whereby each word was removed from the lexicon in turn and its syllabification was inferred from all other words. This procedure was chosen for consistency with previous studies.

Results were computed using word and juncture accuracies. Word accuracy is simply the number of words syllabified by the method in exactly the same way as is given by the standard used (in this case, the Full Lexicon). Juncture accuracy compares syllabification at the sub-word level. Each position between letters (juncture) is assessed to determine whether it was classified correctly. For example, the Italian word “sempre” has five junctures (denoted by a ‘?’) and can be shown as “s?e?m?p?p?e?” The accepted syllabification, according to the Italian lexicon, is “sem|pre”. If an algorithm syllabifies the word as “semp|re”, this is considered entirely wrong in terms of word accuracy, however it is 60% (3/5) correct in terms of juncture accuracy, as shown in Table 4 in which C and I correspond to correctly and incorrectly syllabified junctures, respectively.
Achieving a high word accuracy is therefore more difficult than a high junction accuracy and a better "acid test" for the procedure. Additionally, if, as is typically the case, the syllabification of an entire word is what is required for real-world applications, the algorithm's word accuracy performance must be high. Although this guarantees a high junction accuracy, the converse is not also true.

In addition to applying each individual rule set to the word list, the rules, as a whole, were evaluated using a majority vote system. This was performed both at the word and junction levels. When all three rule sets differed in syllabification at the word level, the boundary placement given by the overall best ranking rule set was chosen. This was possible due to the fact that rule sets had already been evaluated individually. Because junction level syllabification is a binary decision (either a syllable boundary is placed or not), the problem of three differing syllabifications did not arise for this approach.

Ideally, as was done by Marchand et al. (in press), another syllabified lexicon could be used to ascertain whether the results found using the Full Lexicon as the gold standard were accurate. However, such a resource was not available in the Italian language. To compensate for this, and to give the rule-based algorithms the benefit of the doubt, each rule-based algorithm was used as a gold standard and all algorithms were evaluated against these rule-derived syllabifications in turn.

5.2. Results

The results for all automatic syllabification algorithms are presented in Table 5. Although all 15 weight sets were used for the Look-Up Procedure, only the top five are reported. The difference in word accuracy performance between the best rule-based method (SYL-LABE) and the best data-driven method (SbA) is approximately 10%. A Chi-square test ($\chi^2$) reveals $\chi^2_{obt} = 2977.0$ for words and $\chi^2_{obt} = 5030.3$ for junctions. These differences between SbA and SYL-LABE are highly statistically significant ($p < 0.01$ in both word and junction accuracies).

The word errors broken down by the numbers of syllables in the lexical syllabification of the word are given in Figure 6. Regardless of the number of syllables, data-driven algorithms have consistently fewer errors than the rule-based methods (with the exception of Cioni’s Rules for the monosyllabic words).

Figure 7 shows the word errors for each part of speech. Here too, the differences between the data-driven and rule-based algorithms are clear for each type.
Table 5: Syllabification results on the Full Italian Lexicon

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Percentage Correct</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Words</td>
</tr>
<tr>
<td>Cioni’s Rules</td>
<td>86.59</td>
</tr>
<tr>
<td>Hall’s Rules</td>
<td>81.59</td>
</tr>
<tr>
<td>SYL-LABE</td>
<td>89.77</td>
</tr>
<tr>
<td>Majority</td>
<td></td>
</tr>
<tr>
<td>Juncture-based</td>
<td>86.35</td>
</tr>
<tr>
<td>Word-based</td>
<td>84.89</td>
</tr>
<tr>
<td>Syllabification by Analogy</td>
<td></td>
</tr>
<tr>
<td></td>
<td>97.70</td>
</tr>
<tr>
<td>Look-up Procedure</td>
<td></td>
</tr>
<tr>
<td>version 10</td>
<td>96.43</td>
</tr>
<tr>
<td>version 11</td>
<td>96.04</td>
</tr>
<tr>
<td>version 8</td>
<td>96.02</td>
</tr>
<tr>
<td>version 13</td>
<td>95.93</td>
</tr>
<tr>
<td>version 15</td>
<td>95.82</td>
</tr>
</tbody>
</table>

Table 6: Syllabification results using rule-based lexicons as gold standards

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Cioni</th>
<th>Hall</th>
<th>SYL-LABE</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cioni’s Rules</td>
<td>100.00</td>
<td>89.26</td>
<td>92.75</td>
<td>94.00</td>
</tr>
<tr>
<td>Hall’s Rules</td>
<td>89.26</td>
<td>100.00</td>
<td>92.77</td>
<td>94.01</td>
</tr>
<tr>
<td>SYL-LABE</td>
<td>92.75</td>
<td>92.77</td>
<td>100.00</td>
<td>95.17</td>
</tr>
<tr>
<td>SbA</td>
<td>98.10</td>
<td>99.56</td>
<td>98.10</td>
<td>98.59</td>
</tr>
<tr>
<td>Look-up Procedure</td>
<td>97.44</td>
<td>98.68</td>
<td>97.71</td>
<td>97.94</td>
</tr>
</tbody>
</table>

Table 6 shows the performance of all algorithms using the rule sets as alternate gold standards. Clearly, the rule-based algorithm from which the alternate lexicon was created will achieve 100% on that lexicon, so, when computing the average of the results over the three rule-based gold standards, the rule-based algorithms are given a strong advantage. However, as the results indicate, the data-driven algorithms still perform better than the rule-based algorithms on average.

An analysis of the syllabification results on the Full Italian lexicon reveals the accuracy of the juncture classification (syllable boundary or non-syllable boundary) in terms of the character on either side of the juncture (in other words, by the bigrams). These errors can be categorized according to whether the bigram contained consonants or vowels. The percentage of errors attributed to each of the four possible bigram types (CC, CV, VC, and VV) for each syllabification approach is shown in Figure 8. The majority of the errors fall into the VV bigram category. This is interesting to note due to the differences amongst the rule sets for syllabification of VV bigrams.

Discrepancies in performance amongst the rule-based methods are attributed to differences in the rule sets used by each. Although some rules are consistent between methods, others are vastly different (some differences are shown by
example in Table 7). For instance, the rules given by Hall (1974) state that no vowel cluster should ever be separated by a syllable boundary while Cioni (1997) states that when the vowels ‘a’, ‘e’, and ‘o’ are adjacent within a word (e.g. ‘ae’ or ‘eo’), they are not in the same syllable. Simple analysis of the instances of these bigrams in the lexicon reveals that, for bigrams involving the vowels ‘a’, ‘e’, and ‘o’, Hall’s rule is always wrong while Cioni’s rule is always correct. Although, as stated above, the rule sets have a great deal in common, these differences show that they do not always agree.

The fact that the majority of the errors occurs in the VV class is even more noteworthy when the low proportion of these bigrams in the entire data set is taken into account. Indeed, VV bigrams take up only 3.72% of the lexicon, indicating that this is a specific area in which Italian rule sets must be improved. It is interesting to note that vowels have also been found to be a source of a large number of the errors which occur when automatically converting letters.
to phonemes in English (Marchand and Damper, 2000).

The one CC bigram which occurs in the top ten errors of four of the five syllabification procedures is “sc”. This is notable because of the difficulty that this bigram presents to syllabification. Indeed, Bertinetto (2004) presents evidence indicating that this bigram may be entirely undecidable and exhibit neither the traits which typically characterize bigrams occurring together in the same syllable or those which usually have a syllable boundary placed between them.

The difference between the Look-up Procedure error distribution (in Figure 8 the errors reported are those found for version 10) and that of all other approaches (and especially SbA) testifies to the fact that this method of syllabification works entirely differently from the other tested algorithms. We also computed the error rate of each method on the four bigram types (for example, the error rate on the CC bigram is found by dividing the number of errors made by the method in question by the total number of occurrences of this bigram type). Both data-driven algorithms perform much better than all rule sets for both CC and VV bigrams, with the Look-up Procedure doing slightly better.
5.3. Discussion

Although mentioned in the description of the SYL-LABE rules above, it should not be forgotten that stress information is required in order to properly apply these rules. This is especially worth noting because determining the stressed syllable for words of three or more syllables is a difficult task. Indeed, according to Burani and Arduino (2004), it cannot be determined by rule. This may be a key to the superior performance of this rule set over the others. It is interesting to note that the data-driven methods (without stress information of any kind) were still more accurate than SYL-LABE.

Furthermore, word accuracy of all rule-sets remained within the 80% range. In fact, despite some differences in the placement of syllable boundaries, the three approaches agree on the syllabification of 88.47% of the words. Of these, 89.86% are correctly syllabified according to the Full Lexicon.

6. Study 2 — Effect of lexicon size

When comparing rule-based to data-driven approaches, the downside of data-driven methods is the necessity of a syllabified lexicon. In contrast, rule-based methods require only the program necessary to run them. It may not
Table 7: Comparison between lexical and rule-based Italian syllabifications

<table>
<thead>
<tr>
<th>Syllabification</th>
<th>Word</th>
<th>Syllabification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Word</td>
<td>stereoscopio</td>
<td>Syllabification</td>
</tr>
<tr>
<td>Lexicon</td>
<td>st e</td>
<td>r e</td>
</tr>
<tr>
<td>Cioni’s Rules</td>
<td>st e</td>
<td>r e</td>
</tr>
<tr>
<td>Hall’s Rules</td>
<td>st e</td>
<td>r e o</td>
</tr>
<tr>
<td>SYL-LABE</td>
<td>st e</td>
<td>r e</td>
</tr>
<tr>
<td>Word</td>
<td>cuoiame</td>
<td>Syllabification</td>
</tr>
<tr>
<td>Lexicon</td>
<td>cu o</td>
<td>i a</td>
</tr>
<tr>
<td>Cioni’s Rules</td>
<td>cu o</td>
<td>i a</td>
</tr>
<tr>
<td>Hall’s Rules</td>
<td>cu o i a</td>
<td>m e</td>
</tr>
<tr>
<td>SYL-LABE</td>
<td>cu o i a</td>
<td>m e</td>
</tr>
</tbody>
</table>

be worth the increase in word accuracy to use a data-driven method if a large lexicon is required to run it. For this reason, the required lexicon size was tested.

6.1. Method

This was done by dividing the Italian lexicon into 20 partitions of 2236 words. All sets of words shared the same word length distribution. The data-driven algorithms were then run (using the leave-one-out procedure), first using only 5% of the lexicon as training data, but the entire lexicon as test data, and then adding one 5% increment at a time until the entire lexicon was used for training (giving the same results reported in Table 5). In other words, for words in the training set, each word is removed one-by-one from the training data and then syllabified using the results of training with all other words. Then this word is added back into the training set. However, for all other words in the lexicon, training is performed using all words.

6.2. Results

Results are shown in Figure 9. Because juncture accuracy followed the same trend, word accuracy was reported for simplicity. Also, in order to make the results easier to discern, only the best results from the Look-up Procedure are displayed (those obtained using version 10 of the weights).

The Look-up Procedure accuracy increases dramatically between 5% and 10% and also (to a lesser extent) between 10% and 15%. By 20% the data-driven methods are all better than the rule-based ones. The performance of SbA decelerates as the training set increases in a manner similar to that observed for Pronunciation by Analogy when applied to several languages (Soonklang et al., 2008).

6.3. Discussion

It is interesting to note that, even with only 5% of the lexicon as training data (a mere 2236 words), SbA performs better than all three rule-based algorithms.
Figure 9: Comparison rule-set versus data-driven algorithm performance on the Full Italian Lexicon as training data is increased.

With 20% of the lexicon, all data-driven procedures are performing better than the rule-based algorithms.

In order to determine exactly when SbA’s performance exceeds the rule-based accuracy, the 5% portion of the lexicon was further sub-divided into 1% portions (447 words). The results for this are given in Figure 10. From this data, it is clear that SbA surpasses the rule-based algorithms in word accuracy with only 4% of the lexicon (1789 words). The difference between SbA’s word accuracy, when trained on 4% of the lexicon, and the best rule-based algorithm’s results is statistically significant (χ² = 20.7, p < 0.01).

7. Study 3 — Comparison between Italian and English

From the results presented in the two studies described above, the overall performance of all methods appear to be better in Italian than syllabification results in English, previously reported by Marchand et al. (in press). This could be due to the fact that the Full Italian lexicon contained many more words than any of the lexicons used when comparing syllabification methods in English. In this study, it was investigated whether the Italian syllabification results are indeed better than those for English.
7.1. Method

Because the correct syllable boundaries in a word are sometimes disputed, the English comparison used three lexicons: one from Webster’s Pocket Dictionary (19,596 entries), another from the Wordsmyth English Dictionary-Thesaurus (18,016 entries), and a third (called the Overlap English lexicon) which consisted of the 13,594 words with the same syllabification in both of the other lexicons (Marchand et al., in press). To determine whether Italian is indeed easier to syllabify automatically than English, a randomized reduced set of the Full Italian entries was selected. This matched the Overlap lexicon size and, as closely as possible, the word length distribution. The resulting Reduced Italian database also consisted of 13,594 entries.

In order to compare the size effect of the lexicon on the algorithm accuracy between English and Italian, the Reduced Italian Lexicon and the Overlap English Lexicon were divided into 5% portions. Initially, one 5% section was used for training. An additional 5% was added to this to test the data-driven algorithm’s performance at 10%. This was repeated until the training set was the full 100%.

As was done with the Full Lexicon, the Overlap and Reduced Lexicons were further subdivided into 1% portions (136 words) in order to determine more precisely how little of the lexicon is required in order to perform better than the
Table 8: Syllabification results on the Reduced Italian Lexicon

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Percentage Correct</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Words</td>
<td>Junctures</td>
</tr>
<tr>
<td>Cioni’s Rules</td>
<td>87.20</td>
<td>97.03</td>
</tr>
<tr>
<td>Hall’s Rules</td>
<td>80.63</td>
<td>96.16</td>
</tr>
<tr>
<td>SYL-LABE</td>
<td>89.77</td>
<td>97.58</td>
</tr>
<tr>
<td>SbA</td>
<td>95.33</td>
<td>99.09</td>
</tr>
</tbody>
</table>

Table 9: Comparison of syllabification results for Overlap English (Marchand et al., in press) and Reduced Italian lexicons; all the differences were statistically significant ($p < 0.01$)

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Words Correct (%)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Italian</td>
<td>English</td>
</tr>
<tr>
<td>SbA</td>
<td>95.33</td>
<td>85.43</td>
</tr>
<tr>
<td>Look-up Procedure</td>
<td>91.60</td>
<td>73.53</td>
</tr>
<tr>
<td>Rule-based</td>
<td>89.77</td>
<td>36.88</td>
</tr>
</tbody>
</table>

7.2. Results

As shown in Table 8, the performance of the data-driven methods on the Reduced Lexicon was slightly less than on the Full Lexicon, as would be expected given that significantly fewer words were provided for training. However, the difference between the best data-driven algorithm (SbA — 95.33% for words) and best rule-based method (SYL-LABE — 89.77% for words) is still significant ($\chi^2_{obs} = 563.3$, $p < 0.01$).

Table 9 compares the results for the Italian and English languages, showing that, for SbA, the best Look-Up Procedure weights (version 10 for both languages) and the best rule-based methods in each of the two languages, the results obtained for Italian are significantly higher ($p < 0.01$).

The results of the comparison between English and Italian lexicon size effect are shown in Figures 11 and 12. As with the similar figures for the Full Lexicon, only the results of the Look-up Procedure using version 10 of the weights are shown. In spite of the fact that the two languages present different syllabic structures, version 10 not only performs best for Italian but also for English (Marchand et al., in press). This set of weights, slightly unbalanced on the right side (in comparison to version 9), seems to better capture the information relevant to syllabification in these two languages. The two rule-based
algorithms tested for the English lexicon were called Hammond 1 and Hammond 2.

![Graph](image)

Figure 11: Comparison of Overlap English and Reduced Italian Lexicon performance as training data is increased. Here 'I' indicates that the algorithm was applied to the Italian language and 'E' indicates that it was applied to English.

While data-driven algorithms initially perform worse than rule-based algorithms for Italian, this is not so for English. Even with only 1% of the Overlap lexicon, SbA results for English are significantly better than the best rule-based ($\chi^2_{obt} = 160.3$). Look-up Procedure results are significantly better than the best rule based ($\chi^2_{obt} = 76.2, p < 0.001$) with only 2% of the Overlap lexicon (272 words).

7.3. Discussion

It is interesting to note the difference between SbA and Look-up Procedure results. When studying the results shown in Figure 11, the difference decreases with each additional 5% added to the training set for Italian. However, for English, this difference between the two data-driven algorithms remains quite consistent (mean = 14.45%, standard deviation = 0.019).

Comparison between the data-driven algorithms reveals that the Look-up procedure requires the entire Overlap Lexicon to attain word accuracy above 73% while, in contrast, with only 15% of the Overlap Lexicon, SbA already performs at this level. Likewise, for the Reduced Italian Lexicon, the Look-up Procedure requires 95% of the lexicon to attain word accuracy above 91%.
while SbA requires only 25%. These differences within each language, show the strength of SbA over the Look-up Procedure. It is apparent that this method is better able to generalize information from the data provided.

The differences between the two languages also become apparent from these results. While the Look-up Procedure attains word accuracy above 73% only with the entire Overlap Lexicon, the same method only requires 20% of the Reduced Lexicon to surpass this level. For SbA, when syllabifying English, 95% of the lexicon is required to attain word accuracy above 85% but only 5% to perform above this level in Italian. It is clearly much easier to generalize syllabifications from the Italian language than from English.

8. Conclusion

In the words of Hall (1972, page 42) “No set of rules, no matter how extensive and complicated, can ever describe or account for the totality of any individual’s or group’s language.” This statement applies particularly well to the results we have presented.

Previous studies show that data-driven methods outperform rule sets in English syllabification tasks (Marchand et al., in press). The purpose of this study
was to extend the comparison of these two approaches to the syllabification of a language known to have lower syllabic complexity, namely Italian.

The removal of a large number of the words from the original Italian lexicon may have initially been a cause for concern, perhaps simplifying the task. However, in addition to the fact that the vast majority of the words removed were very similar to others within the lexicon (which would make the data-driven task much simpler if left in), these results show that, as the lexicon size is decreased, the task becomes more difficult for the data-driven algorithms. If the difficulty of the task was impacted by the reduced lexicon size, it was to make it more challenging for the data-driven algorithms. In spite of these reductions, these methods still outperform the rule-based approaches.

These results serve to establish data-driven algorithms as methods superior to the tested rule-based approaches for automatically determining syllable boundaries. Furthermore, from a computational perspective, they quantitatively confirm linguistic and psychological findings, which indicate that Italian is simpler and more consistent in syllable structure than English, as stated by Burani et al. (2006); Dauer (1983); Frota and Vigário (2001); Seymour et al. (2003). Although the CV syllable has been found to be the most common in both English and Italian, it constitutes only 34% of the syllables in English (Dauer, 1983) as opposed to a full 60% in Italian (Frota and Vigário, 2001). Such a marked difference should result in Italian being easier to syllabify than English, as the above results have shown to be the case. When the results of Italian syllabification methods are compared to those from English, it is evident that, regardless of the method used, performance on the Italian lexicon is significantly better. This indicates that syllabification must be a more straightforward task in Italian.

In general, these computational approaches (both rule and data-driven methods) add a predictive aspect to the study of syllabification by nature of their ability to delineate syllable boundaries for previously unsyllabified words. This complements structural and behavioral approaches which are typically restricted to words with known syllable boundaries.

These findings open the door to an in depth computational analysis of automatic syllabification over a wider range of languages. Results would shed greater light on the degrees of syllabic complexity amongst languages. Although little has been done toward this for syllabification, similar approaches have proved profitable for letter-to-phoneme conversion (Soonklang et al., 2008; van den Bosch et al., 1994), substantiating, from a computational perspective, the perceived orthographic depths of the European languages of Dutch, English, French, Frisian, German, Norwegian and Spanish.

This comparison on a set of approximately 44,000 Italian words also confirms the superiority of the data-driven methods in terms of both word and juncture accuracies. Overall, all the algorithms presented attain at least 80% word accuracy. The best data-driven method (SbA) reaches a word accuracy of 97.70%, whereas the best rule set (SYL-LABE) achieves an accuracy of 89.77%.

In conclusion, based on the syllabification accuracy of the rule-based and data-driven methods in the Italian language, these results suggest that these rule
sets do not perform as well as data-driven methods for automatic syllabification, even for languages with low syllabic complexity.

Acknowledgements

The authors wish to thank Piero Cosi for providing help with the Italian lexicon. This work was supported in part by funding from the Natural Sciences and Engineering Research Council of Canada (NSERC). In addition, the first author was funded by the National Research Council (NRC) Graduate Student Scholarship Supplement Program (GSSSP), and an Izaak Walton Killam Predoctoral Scholarship. Some of the preliminary data presented here was previously presented at the Sixth International Speech Communication Association (ISCA) Workshop on Speech Synthesis.

References


Bertinetto, P. M., Cioni, L., Agonigi, M., 1994. The hierarchical structure of the syllable in Italian (or: Does one really start from the beginning?). Quaderni del Laboratorio di Linguistica 8, 1–21.


