Tagging Syllable Boundaries With Joint N-Gram Models

Helmut Schmid, Bernd Möbius, Julia Weidenkaff

Institute of Natural Language Processing (IMS), University of Stuttgart, Germany
{schmid,moebius,weidenja}@ims.uni-stuttgart.de

Abstract

This paper presents a statistical method for the segmentation of words into syllables which is based on a joint n-gram model. Our system assigns syllable boundaries to phonetically transcribed words. The syllabification task was formulated as a tagging task. The syllable tagger was trained on syllable-annotated phone sequences. In an evaluation using ten-fold cross-validation, the system correctly predicted the syllabification of German words with an accuracy by word of 99.85%, which clearly exceeds results previously reported in the literature. The best performance was observed for a context size of five preceding phones. A detailed qualitative error analysis suggests that a further reduction of the error rate by up to 90% is possible by eliminating inconsistencies in the training database.

Index Terms: syllabification, joint n-gram models, German

1. Introduction

Automatic syllabification of words is an important task in a number of natural language processing and speech technology applications. Knowledge of syllable structure is crucial for assigning phone durations and F_0 contours as well as selecting appropriate units in concatenative speech synthesis [1]. It has also been shown to improve word modeling in automatic speech recognition [2]. Early approaches to automatic syllabification were knowledge-based and built on the maximum onset principle [3] or the sonority hierarchy [4, 5]. Recently, work on data-driven syllabification has been presented [6, 7, 12].

Syllabification can be performed either on the orthographic representation of words or on the phonetic symbol string (phonetic transcription) of words, depending on the particular task at hand. The former design may be useful under the assumption that knowledge of syllable structure can improve pronunciation [7]. The latter design is sometimes applied in speech synthesis for languages, such as German, in which pronunciation is known to be sensitive to morphological structure and syllabification can be performed more reliably based on the phonetic transcription than on orthography [8].

The syllable structure of German is complex. German phonotactics allows consonant clusters in the onset and coda of syllables. The maximum number of consonants is 3 in the onset (e.g., [StR]) and 5 in the coda (e.g., [mpfst]).[1] Thus, a maximum number of 8 consecutive consonants may occur across syllable boundaries. This complexity of onset and coda structures poses problems for a syllabification algorithm because multiple alternative syllable boundary locations are usually possible in polysyllabic words.

The automatic syllabification method presented in this paper aims at assigning syllable boundaries to phonetically transcribed words. This task is formulated as a tagging task: each phone symbol in the transcription of a word is annotated as either preceding a syllable boundary or not preceding a syllable boundary. Our system learns the probabilities of syllable boundaries from annotated corpora and predicts the locations of syllable boundaries in previously unseen, unsyllabified, transcriptions of words.

In a series of experiments, the performance of the syllable tagger was evaluated on the German part of the CELEX lexical database [9]. The syllable tagger was found to predict correctly the syllable boundaries in words from test data held out from the training set with an accuracy by word of 99.85%.

It is important to note that the syllabification information given by the CELEX database is used in our experiments both as an evidence base from which the syllabification of the test data is inferred, and as a reference syllabification for evaluation. This is an established procedure in the absence of a gold standard [7]. Therefore, the performance of the syllable tagger is a function of, among other factors, the quality of the evidence base. This is not a trivial statement, as the qualitative error analysis in section 3.1 will illustrate.

2. Syllable tagger

We have chosen a tagging approach to syllabification: our syllabification program annotates each phone symbol in the transcription of a word either with a 'B' tag (indicating a syllable boundary after the phone) or an 'N' tag (no syllable boundary). For instance, the correct tagging of the phone sequence [pake:t@] (*Pakete* 'packages, parcels') is 'p/N a/B k/N e:/B t/N @/N' [pa . ke: . t@].

2.1. Statistical model

Our syllabifier chooses the most likely tag sequence $\hat{b}_1^n = \hat{b}_1, \hat{b}_2, ..., \hat{b}_n$ for the given phone sequence p_1^n. In other words, it chooses the tag sequence which maximizes the conditional probability $P(b_1^n|p_1^n)$ according to equation 1. The prior probability $P(p_1^n)$ of the phone sequence in equation 2 is independent of the tag sequence b_1^n. Therefore, it has no influence on the ranking of the different tag sequences and can be ignored, which leads to equation 3. Equation 4 is obtained by decomposing the probability of the tagged phone sequence into a product of conditional probabilities. After introducing a Markov assumption (i.e. we assume that each tag–phone pair only depends on the k preceding tags and phones, and that the probabilities are time-invariant), equation 4 simplifies to equation 5.

$$\begin{aligned} \hat{b}_1^n &= \arg\max_{b_1^n} P(b_1^n|p_1^n) && (1) \\ &= \arg\max_{b_1^n} P(b_1^n, p_1^n)/P(p_1^n) && (2) \\ &= \arg\max_{b_1^n} P(b_1^n, p_1^n) && (3) \end{aligned}$$

[1] SAMPA notation is used for phonetic transcriptions in this paper.

$$= \quad \arg\max_{b_1^n} \prod_{i=1}^{n} P(b_i, p_i | b_1^{i-1}, p_1^{i-1}) \tag{4}$$

$$= \quad \arg\max_{b_1^n} \prod_{i=1}^{n+1} P(b_i, p_i | b_{i-k}^{i-1}, p_{i-k}^{i-1}) \tag{5}$$

In order to make sure that all conditional probabilities in equation 5 are well-defined, we set $p_i = \#$ and $b_i = N$ for $i < 1$. (The dummy phone symbol # is used to indicate the word boundary.) Furthermore, we define $p_{k+1} = \#$ and $b_{k+1} = N$ to mark the end of the word, and we multiply the conditional probabilities from position 1 to $n + 1$ rather than n.

Without the last refinement, the erroneous syllabification [za:k . t] (*sagt* 'says') would be probable, because it follows from equation 5 that the probability of any sequence $\langle b_1^n, p_1^n \rangle$ must be at least as high as the sum of the probabilities of all sequences which start with $\langle b_1^n, p_1^n \rangle$. In our example, [za:k . t] (*sagt*) must be more probable than [za:k . t@] (*sagte*), [za:k . t@st], (*sagtest*), [za:k . t@n] (*sagten*), and [za:k . t@t] (*sagtet*) together, although the last syllable [t] is not a valid German syllable.

Our syllable tagger uses the Viterbi algorithm [10] to efficiently compute the most probable tag sequence according to equation 5.

2.2. Parameter smoothing

The model parameters $P(b, t|C)$ with $C = b_1^k, p_1^k$ need to be smoothed in order to avoid problems with zero probabilities. We used the following simple backoff strategy which smoothes frequencies by adding the backoff probability which is multiplied by some predefined weight parameter Θ. The smoothed frequencies are then divided by the context frequency plus Θ to obtain the probability estimates.

$$P(b, t|C) = \frac{F(C, b, t) + \Theta P(b, t|C')}{F(C) + \Theta}$$

The example in Figure 1 shows how the context C is generalized to the backoff context C'. The initial phone ([E]) of the context, consisting of the phone sequence [Efl] and the pertinent syllable boundary tags, is first generalized to a flag indicating whether the phone was a vowel (+) or a consonant (–), i.e., it is replaced by '+' in the present example. In the second generalization step, this flag is deleted together with the respective syllable boundary tag ('N'), leaving a shorter context phone–tag sequence. The same procedure is iteratively applied, if necessary, to the remaining context, which can be maximally reduced to zero.

```
E/N f/B l/N
+/N f/B l/N
    f/B l/N
    -/B l/N
        l/N
        -/N
      <zero>
```

Figure 1: *Illustration of the backoff strategy for parameter smoothing; see text for details.*

The exact value of the smoothing parameter Θ had little influence on the syllabification accuracy.

2.3. Syllable filter

Initial tests indicated that the syllable tagger described above sometimes produced syllables containing two vowels, as in [trEntSko:ts] (*Trenchcoats*), or without a vowel, like the last syllable in [rau . bau . ts] (*Raubauz* 'brute'). However, vowel-less syllables or syllables comprising two or more vowels violate general constraints on the syllable structure of German, which can be stated by the regular expression C*VC*.[2]

We modified the tagger in order to make sure that the tagger only produces syllables with exactly one vowel. Each state of the new tagger corresponds to a k-tuple of tag–phone pairs (as before) plus an additional flag indicating whether or not the current partial syllable already contains a vowel or not. This flag is cleared in the start state. If a vowel is encountered, the flag is set (i.e., the flag of a state which emits a vowel must be set). After a syllable boundary, the flag is cleared again.[3] If a vowel is immediately followed by a syllable boundary, the flag is also cleared. In all other cases the vowel flag remains unchanged.[4]

Syllables with zero or two vowels are excluded by the following additional restrictions: (i) If the vowel flag is set, the next phone cannot be a vowel (eliminating syllables with two vowels).[5] (ii) If the vowel flag is not set, the next "phone" cannot be the word boundary symbol '#' (eliminating final syllables without a vowel).[6] (iii) If the vowel flag is not set and the next phone is a consonant, the next boundary tag must be 'N' (eliminating non-final syllables without a vowel).[7]

This modification also increased the speed of the syllable tagger due to the smaller search space.

3. Evaluation

We evaluated the syllable tagger on the German part of the CELEX lexical database [9], which contains 309,738 different single-wordform entries.[8] 98.2% of the wordforms had more than one syllable. The average number of syllables was 3.6.

The words in the database were split into 10 subcorpora according to two different designs. In test A, the database entries were randomly assigned to the 10 subcorpora. In test B, all wordforms pertaining to the same lemma were assigned to the same subcorpus; otherwise, the assignment of database entries to subcorpora was again random. The design of test B was motivated by the possibility that for a given inflected wordform in the test set, the syllabifier has seen other, very similar, wordforms of the same lemma in the training set. In test C, we investigated whether stress information improves the performance of the tagger. We used the same division into 10 subcorpora as in test B and we encoded the stress with a pseudo phone ['] which was inserted in front of the vowel of the stressed syllable. The word *sechstausend* (six thousand) with two stress accents was

[2] Note that in actual pronunciation, sonorants can be syllabic in German, for instance as a consequence of schwa deletion in unstressed syllables. However, the CELEX database, which provides the training and test material for our syllabifier, represents canonical transcriptions.

[3] If the last boundary tag is 'B', the vowel flag must be cleared.

[4] The vowel flag of the next state is identical to the vowel flag of the preceding state.

[5] There is no transition from a state whose vowel flag is set to a state whose last phone is a vowel.

[6] There is no transition from a state whose vowel flag is not set, to a state whose last "phone" is the word boundary symbol.

[7] There is no transition from a state whose vowel flag is not set, to a state whose last "phone" is a consonant and whose last tag is 'B'.

[8] Entries for multi-word tokens were deleted and entries differing neither in the phone sequence nor the syllable structure were merged.

	prec.	recall	f-score	acc.
CV	99.94%	99.94%	99.94%	99.85%

Table 1: *Syllable boundary tagging results for test design A. Ten-fold cross-validation using optimal context size and smoothing parameter values ($k = 5, \Theta = 10^{-2}$).*

represented as [z'Ekst'auz@nt] in test C.

In a series of experiments, we used the subcorpora 1–9 for training and subcorpus 10 for testing. The phone context size k and the smoothing parameter Θ were systematically varied.

3.1. Test A

Table 1 shows the best results for the syllable boundary prediction task obtained after ten-fold cross-validation on randomly split subcorpora of CELEX. In terms of absolute numbers, this means that out of the total of 309,738 words, only 465 words are assigned a syllabification that differs from the reference syllabification. In the overwhelming majority of words, there is an accurate match between the syllabification assigned by our syllable tagger and those given by the CELEX database. Precision[9] and recall are identical because the one-vowel-per-syllable constraint prevents the tagger from inserting additional syllables or merging two syllables into one. The number of syllables is therefore always correct. All remaining errors resulted from shifting a syllable boundary over neighboring consonants.

The results of this experiment indicate that a context size of less than 4 preceding phones produces a suboptimal, yet still very good, syllabification performance. On the other hand, relatively large contexts (5 or 6 phones) are fairly reliable, despite the fact that data sparsity increases with context size. Evidently, the backoff strategy described in section 2.2 effectively alleviates the sparse data problem. The results obtained with small values of the smoothing parameter are close to optimal. The probability mass assigned to unobserved events is actually vanishingly small. The best context size was 5, but a context size of 4 or 6 is almost equally good.

We manually inspected the first 97 words for which the syllabification assigned by the syllabifier differed from the one provided by the CELEX database. The following error types were identified:

- 48 words were probably mistagged because of inconsistencies in the CELEX database.
- 1 error was caused by a phone error in CELEX.
- In 38 cases, the syllabification probably failed because of missing glottal stop information.[10]
- 10 differences to the reference syllabification were definitely syllabification errors. Examples are the word *Genugtuungen* 'satisfactions' [g@ . nu:k . tu: . U . N@n], which was syllabified as [g@ . nu:k . tu: . UN . @n]; and the word *verfälschten* 'falsified' [fEr . fElS . t@n], which was syllabified as [fEr . fEl . St@n]. 4 out of these 10 errors occurred with foreign words, such as *Ragtime* [r{g . taIm], which was syllabified as [r{ . gtaIm]. This error was probably caused by a sparse data problem: The vowel "{" only occurred in six lemmata.

	prec.	recall	f-score	acc.
CV	99.32%	99.32%	99.32%	98.26%

Table 2: *Results for test B, wordforms of the same lemma in the same subcorpus. Ten-fold cross-validation, $k = 4$, $\Theta = 10^{-4}$.*

	prec.	recall	f-score	acc.
CV	99.32%	99.31%	99.31%	98.23%

Table 3: *Results for test C using stress information. Ten-fold cross-validation, $k = 4$, $\Theta = 10^{-5}$.*

From this error analysis, we conclude that a further reduction of the error rate by up to 90% is possible by eliminating inconsistencies in the CELEX data and adding information about glottal stops.

3.2. Test B

Table 2 shows the results obtained for test design B in which all wordforms pertaining to the same lemma were assigned to the same subcorpus; otherwise, database entries were randomly assigned to the 10 subcorpora. The best context size and the best smoothing parameter value observed in an evaluation on subcorpus 10 were $k = 4$ and $\Theta = 10^{-4}$, respectively. Table 2 reports the results in terms of precision, recall, f-score, and syllabification accuracy by word for the ten-fold cross-validation using these optimal parameter values. The slight drop in performance suggests that in test A the syllabifier capitalized indeed on the random distribution of wordforms of the same lemma across subcorpora.

3.3. Test C

In test C, the tagger had access to information about the stress which was added as a pseudo phone in front of the vowel of the stressed syllable. Otherwise, test C was identical to test B. We tested this version using subcorpora 1–9 as training data and subcorpus 10 as test data. For very small contexts ($k = 2$), the stress information improved the tagger performance by about 0.3%, but for larger contexts ($k > 2$), the accuracy was up to 0.2% lower. Table 3 shows the results obtained with ten-fold cross validation using the optimal parameters from the evaluation on subcorpus 10. These results are almost identical to the results of test B. We conclude that the stress information is not relevant for the syllabification task.

4. Discussion

Syllable boundary prediction in German has been extensively studied by [11]. She experimented with probabilistic context-free grammars and multivariate clustering models. Her best system, evaluated on the CELEX data, achieved 96.88% word accuracy. We evaluated our system also on her data (which was easier than our data) and obtained a word accuracy of 99.98%.

The work most similar to our approach was done by [13] and used a standard trigram POS tagger [14]. The boundary tag set was more complex than our binary tag set, but the context was limited to two preceding phones. The system was evalu-

[9] Precision is the number of correctly predicted syllables devided by the total number of predicted syllables. Recall is the number of correctly predicted syllables divided by the total number of correct syllables. And the f-score is the harmonic mean of precision and recall.

[10] CELEX transcriptions do not include glottal stops, because it is assumed that glottal stops can be assigned by post-lexical rules.

ated on CELEX data and yielded a tagging accuracy of 98.34%. This result is very close to the word accuracy of 98.32% of our syllable tagger for the same context size of $k = 2$, but it is not clear whether the percentage of correctly syllabified words was reported or the percentage of correctly assigned boundary tags. The latter task should be simpler than the correct syllabification of complete words.

Accuracies higher than 98% are unusual in linguistic processing. Why does the syllable tagger perform so extremely well? One possible reason is that German is a language with rich inflectional paradigms and productive word formation processes. In any text corpus as well as in the wordform database of CELEX, nominal, adjectival and verbal stems occur with different inflectional suffixes and as the bases for many different derivations and compounds. So, for many words in the test data, there was a word with the same base form, but a different inflection, derivation or composition, in the training data. Similarly, for any inflectional or derivational affixes occurring in the test data, there is usually a number of occurrences of base words with the same affix in the training data. The syllable tagger evidently combines these information sources to deduce the correct syllabification of the word.

This consideration was partly taken into account in the test B part of the evaluation, in which all inflectional wordforms of a given lemma were forced into the same subcorpus. This experimental design assured that during the ten-fold cross-validation the syllable tagger was prevented from seeing wordforms in the training set that are just inflectional variants of the test word at hand. Indeed, the slight drop in performance observed as a consequence of this design, relative to the results of the experiments on randomly split subcorpora, suggests that structural information obtained from closely related wordforms constitutes a valuable source of information for the syllable tagger. A more radical test design aiming at keeping all morphologically related wordforms in the same bin, was impractical for the present set of experiments. It is also debatable whether such a design is really desirable: after all, exploiting structural properties and parallelisms between morphologically related wordforms is at the heart of the work on supervised learning of syllabification as it is presented in this paper.

The size of the context window has a large influence on the accuracy of the system. Our results indicate that a context size of 5 preceding phones produces optimal syllabification results. A context size of 4 phones is sufficiently informative to obtain nearly optimal results, and a larger context (6 phones) also yields a nearly optimal performance despite an increased sparsity of training data. However, a context of less than 4 preceding phones produces a suboptimal syllabification performance. The optimal context size can be interpreted with respect to the syllable structure of German. The average syllable length in terms of the number of phones per syllable is 3.73 in the German wordform database of CELEX. This means that for a context size of 4 phones, there is a fair chance of seeing all previous phones of the current syllable. For a context size of 5, the probability is quite high that the last phone of the preceding syllable is also seen, which entails that the preceding syllable boundary is included in the context. Thus, the experimentally determined optimal context size is compatible with an approximate coverage of complete syllables by the training data.

5. Conclusions

We presented a probabilistic approach to automatic syllabification which assigns syllable boundaries to phonetically transcribed words. The syllabification task was formulated as a tagging task. The syllable boundary tagger is based on a joint n-gram model. It was trained on syllable-annotated phone sequences drawn from the German CELEX wordform database. Using ten-fold cross-validation, the syllable tagger correctly predicted the syllabification of words with an accuracy by word of 99.85%. The joint n-gram model requires a context of at least four preceding phones to achieve a close to optimal performance; the best performance was observed for a context size of five preceding phones. The syllabification performance clearly exceeds results previously reported in the literature.

6. References

[1] Sproat, R., Ed., Multilingual Text-to-Speech Synthesis: The Bell Labs Approach, Kluwer, 1998.

[2] Greenberg, S., "Speaking in shorthand—a syllable-centric perspective for understanding pronunciation variation", Speech Comm., 29(2–4):159–176, 1999.

[3] Kahn, D., "Syllable-based generalizations in English phonology", Ph.D. diss., MIT, 1976.

[4] Clements, G.N. "The role of the sonority cycle in core syllabification", in Papers in Laboratory Phonology I: Between the Grammar and Physics of Speech, Kingston, J., Beckman, M.E., Eds., Cambridge UP, 283–333, 1990.

[5] Blevins, J., "The syllable in phonological theory", in The Handbook of Phonological Theory, Goldsmith, J.A., Ed., Blackwell, 206–244, 1995.

[6] Daelemans, W. and van den Bosch, A., "Generalisation performance of back-propagation learning on a syllabification task", in TWLT3: Connectionism and Nat. Lang. Process., Drossaers, M. and Nijholt, A., Eds., Twente Univ. Enschede, 27–37, 1992.

[7] Marchand, Y. and Damper, R.I., "Can syllabification improve pronunciation by analogy of English?", J. Nat. Lang. Engin., 1(1):1–25, 2006.

[8] Möbius, B., "The Bell Labs German text-to-speech system", Comp. Speech Lang., 13:319–358, 1999.

[9] Baayen, R., Piepenbrock, R. and Gulikers, L., "The CELEX lexical database (release 2)", Linguistic Data Consortium, Univ. Pennsylvania, 1995.

[10] Manning, C.D. and Schütze, H., Foundations of Statistical Natural Language Processing, MIT Press, 1999.

[11] Müller, K., "Automatic detection of syllable boundaries combining the advantages of treebank and bracketed corpora training", Proc. 39th Ann. Meet. ACL, Toulouse, 402–409, 2001.

[12] Kiraz, G.A. and Möbius, B., "Multilingual syllabification using weighted finite-state transducers", Proc. Third Internat. Workshop on Speech Synthesis, Jenolan Caves, Australia, 71–76, 1998.

[13] Krenn, B., "Tagging syllables", Proc. Eurospeech-97, Rhodes, Greece, 991–994, 1997.

[14] Brants, T., "TnT - a statistical part-of-speech tagger", Proc. Sixth Appl. Nat. Lang. Process. Conf. ANLP-2000, Seattle, WA, 2000.