Using Morphology and Phoneme History to improve Grapheme-to-Phoneme Conversion

Uwe D. Reichel, Florian Schiel

Department of Phonetics and Speech Communication
University of Munich, Schellingstr. 3, 80799 Munich, Germany
{reichelu, schiel}@phonetik.uni-muenchen.de

Abstract

In this study four statistical grapheme-to-phoneme (G2P) conversion methods for canonical German are compared. The G2P models differ in terms of usage of morphologic information and of phoneme history (left context) information.

In order to evaluate our models we introduce two measures, namely mean normalized Levenshtein distance for classification accuracy and conditional relative entropy for validation of phonotactic smoothness. The results show that morphologic information significantly improves G2P conversion and together with phoneme history leads to a better approximation of the original phonotactics.

Furthermore with the benefit of morphology our models significantly outperform two well established G2P systems.

1. Introduction

As already shown (e.g. [1]), G2P conversion can be improved by integrating morphologic information. The influence of morphology is manifested directly or via syllabic structuring (the latter being used e.g. in [2]). Direct influence is given when phoneme identity is determined by the morpheme class as in following examples (German SAMPA is used for transcriptions):

- *er* in *Erlöser* (*redeemer*) with the segmentation er_{prefix} + $lös_{verb}$ + er_{suffix}. While *er* becomes */QE6/* in the prefix, it has to be mapped on */6/* in the suffix thus yielding the transcription */QE6l2:z6/*.
- *e* in *geben* (*to give*), segmented as geb_{verb} + en_{infl}. The occurrence of */@/* is forbidden in a verb stem containing just one vowel, but obligatory within the inflectional ending *en*. The resulting transcription is therefore */ge:b@n/*.

Indirect influence of morphology via syllable segmentation occurs where morphologic structure determines syllable structure and therefore also phoneme identity:

- *ng* in *Angel* (*fishing rod*) vs. *Angelegenheit* (*affair*): $angel_{noun}$ vs. an_{prefix}+ge_{prefix}+leg_{verb}+en_{suffix}+$heit_{suffix}$. While in *Angel ng* is melted to ambisyllabic */N/*, the sequence of the two prefixes requires an intervening syllable boundary and therefore a separate realization as */ng/* (*/QaN@l/* vs. */Qang@le:g@nhaIt/*).
- *losen* (*to draw lots*) vs. *Losentscheid* (*decision via lot drawing*): los_{verb}+en_{infl} vs. los_{noun}+ent_{prefix}+$scheid_{verb}$. Again the prefix requires a preceding syllable boundary which leads to terminal devoicing and the insertion of a glottal stop (*/lo:z@n/* vs. */lo:sQEntSaIt/*).

Because including morphologic information in G2P systems via hand-crafted rules is very time consuming, we preferred a statistical approach, where morphologic information and phoneme history are treated as features for G2P classification in a supervised learning framework. See e.g. [3] for an overview over some machine learning approaches to G2P conversion.

2. Morphological Segmentation

2.1. BALLOON Architecture

BALLOON is a toolkit for lexicon creation, which has been developed at our department within the BAS (Bavarian Archive for Speech Signals) project in 2004. It derives automatically from part of speech labelled text: morphological segmentation, orthographical syllable segmentation, G2P conversion, phonologic syllable segmentation and lexical stress. A more detailed introduction will be published in the 2005 working papers of our department (FIPKM). Morphological segmentation is used for orthographic syllable segmentation and G2P conversion (together with syllable segmentation). Syllable segmentation and G2P conversion are carried out by C4.5 decision trees [4].

2.2. Segmentation Algorithm

The rule based segmentation algorithm (see [5]) works roughly in the following way: Each type w of the input text is recursively divided into string prefixes and suffixes from left to right until a permitted segmentation is achieved or until the end of w is reached. In the course of the recursion a boundary dividing the current string in prefix and suffix is accepted if (i) the prefix is found in the morpheme lexicon, (ii) there exists a permitted segmentation for the suffix or (if not) the suffix is found in the lexicon, (iii) the sequence 'prefix class + class of the first suffix segment' is not in conflict with German morphotactics and (iv) the class of the last suffix is in correspondence with w's part of speech.

If a substring does not occur in the lexicon, a context dependent generation of possible lemmas is carried out in order to check if it is an allomorphic variant of some morpheme entry.

Morphological entries are taken partly from the morpheme lexicon and partly derived automatically from the input text applying stemming and allomorph generation. The morpheme lexicon itself has been constructed that way and corrected by hand. At present it comprises 10715 entries each composed by a morph and its morphologic class.

2.3. Results of Morphologic Analysis

The algorithm's performance was evaluated for 2000 word types chosen randomly from those which in principle are segmentable due to their part of speech. The average number of morphemes per word was 2.67. Omissions and false insertions of segment boundaries were counted, a boundary displacement was punished by adding one omission and one insertion.

For evaluation of the morpheme classification a mismatch between original and predicted morpheme class was treated as an error, if one of the three following requirements was violated:

- the right class is obtainable via forced backtracking,
- the exchange of the classes has no impact on syllabification,
- both classes belong either to lexical or grammatical morphemes.

Following these guidelines the morphological analysis yielded the results shown in Table 1. *Classification accuracy* accounts for the percentage of words with completely correct segmentation and classification, *segmentation accuracy* for the percentage with correct segmentation. *Recall* and *precision* are given for the retrieval of segment boundaries.

Table 1: *Results of the Morphological Analysis (in %)*

classification accuracy	91.60
segmentation accuracy	91.65
recall	95.05
precision	97.75

Problematic cases Two examples of erroneous morphological analyses directly affecting G2P conversion are given here.

- *Handballergebnis* (*handball result*) was analyzed the following way (only the relevant morpheme class is shown): *hand+ball+er*$_{suffix}$*+geb+nis*. *er* being actually a prefix in ***Er**gebnis* is incorrectly classified as a suffix of *Handball**er*** (*handball player*). This wrong classification leads to a wrong conversion of *er* to */6/* instead of */QE6/*. Such fatal confusions of prefixes and suffixes can occur wherever they are interchangeable according to morphotactics.
- Due to a part of speech tagging error, *kontrastreich* was considered as a noun with the meaning *"contra trick"* or *"empire of contrast"* depending on its segmentation, instead of an adjective meaning *"rich in contrast"*. The morphologic segmentation algorithm as described above proceeding from left to right first delivers the trick alternative *kontra*$_{prefix}$+*streich*$_{noun}$. The *s* is therefore wrongly mapped on an */S/* instead of an */s/* as it should be the case in the right segmentation *kontrast*$_{noun}$ + *reich*$_{suffix}$.

In both cases forced backtracking would lead to a correct morphological analysis, and in the second case the given analysis is not even wrong, but just less probable. A future task would thus be to provide probability information about morpheme sequences for the morphologic analyzer.

3. Syllable Segmentation and Grapheme-to-Phoneme Models

In our approach G2P conversion is a one-to-one mapping from the set of graphemes to the set of phonemes (German SAMPA). To cope with any n-to-n relation the phoneme set also comprises the empty phoneme as well as phoneme clusters.

Four statistical decision tree models are constructed from different pools of automatically extracted features. $M1$ uses neither morphological information nor the phoneme history, $M2$ uses phoneme history (of length 3), $M3$ morphological information, and $M4$ both. All models use grapheme context and syllable specifications derived from the output of the syllable models that predict for each grapheme whether a strong, an ambisyllabic or no boundary follows. The syllable model $S1$ underlying $M1$ and $M2$ is trained without the use of morphology, whereas model $S2$ supplying information to $M3$ and $M4$ incorporates morphology.

3.1. Features

Tables 2 and 3 list the features on which these models were trained.

Table 2: *Feature Pool for Orthographic Syllable Segmentation Models S1, S2; see text for explications*

	S1	S2
GRAPHEME	+	+
MORPH_BOUND	–	+
MORPH_CLASS	–	+
number of features	8	14
number of nodes	1733	722

Table 3: *Feature Pool for Grapheme-to-Phoneme models M1–M4; see text for explications*

	M1	M2	M3	M4
GRAPHEME	+	+	+	+
MORPH_BOUND	–	–	+	+
MORPH_CLASS	–	–	+	+
SYLL_BOUND	+	+	+	+
SYLL_SPECS	+	+	+	+
POS_IN_SYLL	+	+	+	+
PHONHIST	–	+	–	+
number of features	15	14	11	16
number of nodes	3211	2783	2617	2957

GRAPHEME stands for the current letter as well as for an abstract classification of this letter in consonants and vowels.

MORPH_BOUND encodes whether a syllable relevant morpheme boundary, a non-relevant boundary or no boundary follows the current letter. Morphologic boundaries relevant for syllabification occur in front of:

- all morphemes that do not belong to inflectional endings (INFL), suffixes (SFX), linking morphemes and comparison morphemes
- INFL, SFX with initial consonant and syllable nucleus
- INFL, SFX with initial vowel if the preceding morpheme ends with a vowel

Often an irrelevant morpheme boundary affects syllabification in a certain window around this boundary, like the noun suffix *ung* that in post-consonantal position co-occurs with a syllable boundary in front of the preceeding consonant (*Be+wäh+rung*; *probation*). This phenomenon is accounted for by windowing (see below).

MORPH_CLASS subsumes two features: the class of the morpheme the regarded grapheme belongs to, and a more abstract morpheme classification in lexical and grammatical morphemes.

SYLL_BOUND specifies whether a syllable boundary follows or not, further dividing between presence and absence of ambisyllabicity.

SYLL_SPECS characterizes the syllable onset (naked vs. covered) and offset (open vs. closed vs. ambisyllabic).

POS_IN_SYLL reflects the position of the current grapheme within its syllable. Possible locations are head, nucleus, coda and, in case of ambisyllabicity, juncture.

Beside POS_IN_SYLL, SYLL_SPECS and the phoneme history PHONHIST all features were extracted within a symmetric window of length 7 centered on the current grapheme.

3.2. Model Development

In order to find well performing and small models an incremental combination of features was carried out during the development phase. An initially empty feature list *al* is filled iteratively with attributes. At each iteration step among all attributes *ral* that one (*ba*) is chosen, which together with *al* leads to the best performing model when being applied on the development data. *ba* is then removed from *ral* and added to *al*, which is stored in a *best_al* table together with its performance. When *ral* is empty the best performing feature combination of *best_al* is used for application on the test data. This incremental method led in all cases to slightly better results than applying the decision tree on all features at once. In one case this difference was significant (two-tailed McNemar test, $p = 0.01$), in three cases significance was slightly missed. The number of features finally used by the models as well as the size of the resulting decision trees (in nodes) are given in the bottom lines of Tables 2 and 3.

4. Data

The data for syllable segmentation consists of 12073 word types automatically divided into syllables followed by manual corrections.

The data for G2P conversion was taken from the hand corrected Phonolex_core canonical pronunciation dictionary available at our department and has been aligned automatically. Alignments identified as erroneous by some heuristic postprocessing, among them mainly foreign language material, were discarded. In total, 18412 entries remained for our study.

In both the syllable and the G2P case 65 % of the data was used for training, 22 % for development and 13 % as test set.

5. Evaluation Measures

In the following, we will introduce two evaluation measures for G2P conversion capturing conversion accuracy and preservation of phonotactics: mean normalized Levenshtein distance and conditional relative entropy.

5.1. Mean Normalized Levenshtein Distance (MNLD)

As evaluation measures both word error rate and phoneme error rate have some shortcomings. Computing the word error means all-or-nothing judgments for each word type without the ability to differentiate between different severities of word type errors. Calculating the phoneme error rate in terms of accomplishing a simple phoneme by phoneme comparison leads to erroneous 'wrong' judgments for the remainder of the phoneme string after omissions and insertions.

Therefore we use the normalized Levenshtein distance of two strings, which is defined here as the minimum number of edit operations (the Levenshtein distance) to convert one string into the other divided by the length of the reference string. The original transcription serves as reference for comparison with the model's output. The Levenshtein distance was determined using the Wagner-Fisher algorithm [6]. In our case edit operations consist of insertions, deletions and substitutions. Finally the mean of all normalized distances was calculated.

5.2. Conditional Relative Entropy (CRE)

Phonotactics of a language can be expressed in terms of conditional probabilities of a phoneme y given a phoneme history x. In order to measure the phonotactic divergence of two pronunciation corpora represented by such conditional probability distributions we used the CRE measure given in the following equation.

$$\text{CRE} = D(p(y|x)\|q(y|x)) = \sum_x p(x) \sum_y p(y|x) \log \frac{p(y|x)}{q(y|x)}$$

p and q are the conditional probabilities of phoneme y given the phoneme history x. p is derived from the original data and q from the output of our G2P models. The relative entropy $D(p\|q)$ is a measure of the divergence of the probability distributions p and q expressed in the average number of extra bits needed to encode events from p taking a code based on q. The lower thus the entropy values, the more similar two phonotactic systems.

A G2P device designed to approach original phonotactics as close as possible should produce an output with a low CRE value when being compared with original data.

To evaluate the adequacy of this entropy measure we compared the divergences of a reference dictionary with two other pronunciation dictionaries of which could be stated, that their phonotactics deviate in different amounts from that of the reference one. As reference the Phonolex_core dictionary (PD) was used, as candidates a sub-corpus of the Hadi-Bomp Dictionary (HD) available at the IKP, University of Bonn <http://www.ikp.uni-bonn.de/dt/forsch/phonetik/bomp>, and of the CMU Pronounciation Dictionary (Version 0.6, CD), provided by the Carnegie Mellon University, Pittsburgh <http://www.speech.cs.cmu.edu/cgi-bin/cmudict>. Since both PD and HD are German dictionaries, while CD is for American English, it can be assumed that phonotactics of HD and PD are more similar than those of CD and PD. Pairwise disjunct samples of the same size were taken for entropy calculation after the three phoneme sets had been mapped on a uniform German set.

As a result the comparison of HD and PD yields a much lower trigram CRE value than the comparison of CD and PD (see Table 4) which reflects well the different amount of phonotactic similarity. We regard this as an indication of the CRE's general adequacy for measurement of phonotactic similarity.

Table 4: *Conditional relative entropy measures for phoneme trigrams in different pronunciation dictionaries*

	Phonolex_core
Hadi-Bomp	3.072
CMU P. Dict.	4.162

6. Results

6.1. Syllable Segmentation

The syllable models predict for each letter whether a syllable boundary follows or it is part of a ambisyllabic juncture or no boundary follows. The models were evaluated on the same test data. Results for $S1$ (without morphology) and $S2$ (with morphology) are shown in Table 5. $S2$ performs significantly better than $S1$ (two-tailed McNemar test for dichotomous variables in related samples, $p = 0.001$).

Table 5: *Results for Syllable Segmentation (in %)*

	letter error rate	word error rate
S1	2.40	18.60
S2	1.10	7.52

6.2. Grapheme-to-Phoneme Conversion

All four models were evaluated on the same test set. For practical reasons clusters consisting of a vowel symbol with a preceding glottal stop symbol and/or a following lengthening marker were merged to one character during the calculation of the Levenshtein distances.

The MNLDs diverged significantly (Friedman test for related samples, $p = 0.01$; this non-parametric test had to be chosen because the requirements for parametric tests were not met by our data). Pairwise comparison of the MNLDs revealed that the incorporation of morphological information lead to significantly lower MNLDs ($p = 0.05$) while the improvement connected to the use of phoneme history was not significant (two-tailed Wilcoxon and Wilcox test of multiple comparisons of related samples). The same findings were obtained by pairwise comparison of word error rates (two-tailed McNemar test).

But as can be seen in Figure 1, the use of phoneme history always caused a reduction of trigram CRE compared with the respective counterpart model (M2 vs. M1, M4 vs. M3) and therefore an improved adaption to the original phonotactics.

We also compared our models with P-TRA (PT), a rule based G2P algorithm ([7], further developed in [2]) and the purely data driven approach of Daelemans and van den Bosch [8] (DB), the latter trained on our training and development data and both tested on our test set. Concerning MNLD and word error rate (WER) M3 and M4 significantly outperform both of them (two-tailed Wilcoxon and Wilcox test, $p = 0.05$, resp. two-tailed McNemar test, $p = 0.001$).

Table 6: *Results for Grapheme-to-Phoneme Conversion*

	phon. hist.	morph.	WER	MNLD	CRE
M1	–	–	23.54	0.041	1.081
M2	+	–	21.77	0.039	0.973
M3	–	+	16.17	0.027	0.771
M4	+	+	15.12	0.026	0.700
PT	+	-	23.64	0.038	1.141
DB	+	-	20.72	0.033	0.837

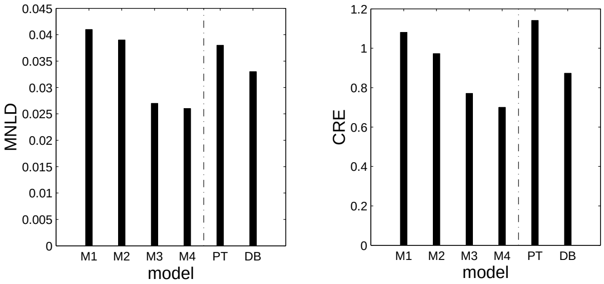

Figure 1: *Mean Normalized Levenshtein Distances (left) and Conditional Relative Entropies between the models' outputs and the test data (right).*

7. Conclusions

We enriched a machine learning approach to German G2P conversion with morphologic analysis and phoneme history. In order to evaluate the influence of these two information sources we introduced two evaluation measures: MNLD and CRE. MNLD was used to measure classification accuracy. It allows a finer grained evaluation than word error rate. A preliminary examination of CRE indicated its appropriateness for evaluating the phonotactic quality of a G2P output.

The application of these measures revealed significant improvements of G2P performance when morphologic information is included, and a higher phonotactic quality by integration of morphology and phoneme history.

The fact that the morphology based models also outperform the well established rule based G2P system P-TRA and the data driven model of Daelemans and van den Bosch, supports our hybrid linguistic and statistic approach to G2P conversion.

8. Acknowledgments

We thank Hartmut Pfitzinger for providing us with his G2P aligner and his implementation of the algorithm of Daelemans and van den Bosch.

9. References

[1] K. Wothke, "Morphologically based automatic phonetic transcription," *IBM Systems Journal*, vol. 32, no. 3, 1993.

[2] M. Libossek and F. Schiel, "Syllable-based Text-to-Phoneme Conversion for German," in *Proc. ICSLP*, Beijing, 2000.

[3] F. Yvon, "Self-learning techniques for grapheme-to-phoneme conversion," Onomastica Research Colloquium, London, 1994.

[4] J. R. Quinlan, *C4.5: Programs for Machine Learning*. San Mateo: Morgan Kaufmann, 1993.

[5] U. D. Reichel and K. Weilhammer, "Automated Morphological Segmentation and Evaluation," in *Proc. LREC*, Lisbon, 2004.

[6] R. Wagner and M. Fischer, "The string to string correction problem," *Journal of the Association for Computing Machinery*, vol. 21, no. 1, 1974.

[7] D. Stock, "P-TRA – Eine Programmiersprache zur phonetischen Transkription," in *Beiträge zur Angewandten und Experimentellen Phonetik*, W. Hess and W. Sendlmeier, Eds. Stuttgart: Franz Steiner Verlag, 1992.

[8] W. Daelemans and A. van den Bosch, "Language-Independent Data-Oriented Grapheme-to-Phoneme Conversion," in *Progress in Speech Synthesis*, J. P. H. e. a. van Santen, Ed. New York: Springer, 1997.