Using Machine Learning Techniques for Grapheme to Phoneme Transcription

Franco Mana, Paolo Massimino, Alberto Pacchiotti

Voice Technology
Loquendo, Vocal Technology and Services
Turin, Italy
{franco.mana,paolo.massimino,alberto.pacchiotti}@loquendo.com

Abstract

The renewed interest in grapheme to phoneme conversion (G2P), due to the need of developing multilingual speech synthesizers and recognizers, suggests new approaches more efficient than the traditional rule&exception ones. A number of studies have been performed to investigate the possible use of machine learning techniques to extract phonetic knowledge in a automatic way starting from a lexicon.
In this paper, we present the results of our experiments in this research field. Starting from the state of art, our contribution is in the development of a language-independent learning scheme for G2P based on Classification and Regression Trees (CART). To validate our approach, we realized G2P converters for the following languages: British English, American English, French and Brazilian Portuguese.

1. Introduction

Phonetic transcription, as an intermediate representation between written text and speech, has always had a crucial role in text-to-speech (TTS) and automatic speech recognition (ASR). Phonemes, together with prosodic information, are the input to the voice production phase in speech synthesis. Likewise, in speech recognition systems based on open vocabulary, where on-the-fly vocabularies can be generated at run-time from graphemic word forms, the phonetic transcription of the words is the starting point to create the acoustical models used by the recognizer.

Obtaining the correct phoneme string from written text is not a trivial task, and for some languages such as English, it's a challenging one. Nevertheless, Grapheme to Phoneme conversion (G2P) has been faced from the very beginning of speech technology research and any existing speech system implements its own language-dependent solution. What is bringing renewed interest on the subject is the recent move towards multilingual systems.

Systems that should be able to deal with several languages should preferably rely on language-independent engines acting on distinct linguistic knowledge bases. Moreover, the time needed to develop a new language should be reduced as far as possible, overcoming the difficulty of acquiring a deep insight into the language-specific phonetic features. This is why automatic language-independent approaches to G2P conversion are highly desirable.

Some work in this direction has already been reported in the literature [1,2,3,4,5]. Our own contribution was motivated by the need of providing an efficient environment to implement G2P converters for the different languages developed for Actor™ [6] and Flexus™ [7,8], respectively the text-to-speech and the speech recognition systems realized at Loquendo (once CSELT).

Our work is still in progress, but some promising results have been already obtained and exploited in the current versions of the systems, as reported in the following.

2. The Grapheme to Phoneme Task

In general, the G2P task amounts to converting a grapheme stream into a phonetic one. When embedded in a TTS system, it is only one of the steps in the conversion of a text into a suitable representation for the synthesizer. Text is preprocessed in order to gain knowledge about its syntactic structure, expand numbers and acronyms, delimit words and locate their lexical stress. For some languages, also morphemes or components of compound words must be identified. Each grapheme word (or morpheme) is then converted into a corresponding phoneme string. After that, further rules are applied accounting for allophonic changes at word boundaries. In ASR systems, G2P is generally confined to the task of providing a reference transcription for the words in the recognizer vocabulary. Both in its TTS and ASR applications, the core of G2P conversion is the transcription of a single grapheme word.

The traditional options for word G2P conversion are explicit rules or lexicon look-up or a combination of the two. Rules would rewrite single graphemes or grapheme clusters into the corresponding phonemes, depending on their context. A phonetician or a computational linguist is required to design the rules, through a time-consuming and knowledge-intensive effort. While for highly regular languages this task is feasible [10], for others, such as English, huge lexica are necessary to account for exceptions. When a high-coverage phonetically transcribed lexicon of inflected forms is available, the alternative is to obtain phonetic transcription by lexicon look-up. In this case, no deep knowledge about the language is required, but ad-hoc implementation in storing and accessing the lexicon is needed to meet real-time constraints. The disadvantage of this choice is that the system has no generalization capability: any word outside the lexicon cannot be transcribed.

A natural way to achieve the constraints of generalization while compacting lexical information, is to extract transcription rules form the lexicon in an automatic way using some machine learning techniques.

3. A Machine Learning Approach

As reported in the literature, some studies have been performed in the direction of extracting the transcription rules in an automatic way by machine learning techniques. In particular, inductive learning techniques are able to find out

the common characteristics in the data and generalize them. The machine learning technique we investigated, is our implementation of Classification And Regression Trees (CART), which has already proven useful in predicting prosody for TTS [9]. CART is a well studied and applied technique able to handle both regression and classification tasks. The extracted knowledge is represented into a binary tree and this kind of representation makes it appealing when we have to integrate the G2P system into real time applications. In fact, rules are stored in a compact format, saving storage memory, and the execution time needed to perform a run is low.

Our CART based G2P system (G2P-Cart) is pictured in Figure 1. A sliding grapheme window moves over the word. The window takes into account a subsequence of the word including a focus (the central grapheme to be transcribed) and two non-symmetric contexts, the left and right context of the focus, respectively. For each window the G2P-Cart system is used to produce the phoneme associated to the window focus.

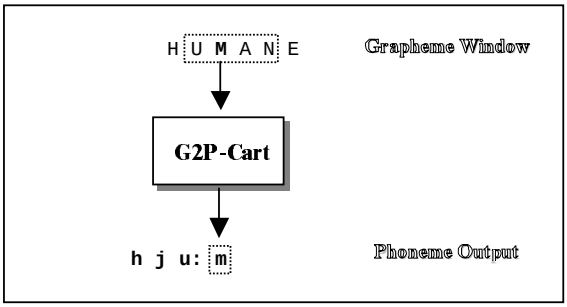

Figure 1. Schematic diagram of the run-time G2P-Cart

The training of the G2P-Cart system starts from a database extracted from the lexicon amounting to a set of pairs <focus & grapheme window, phoneme output>. To obtain such pairs, for each word of the lexicon, we need to align the graphemic string with the phonetic one. For example, the word *humane*, transcribed as [h j`u:m'eIn], is aligned as:

Grapheme:	H	U	-	M	A	N	E
Phoneme:	h	j	u:	m	`eI	n	-

The symbol "-" stays for the NULL grapheme or phoneme. If it appears in the grapheme string a phoneme insertion has occurred. We define a phoneme cluster (U → j+u:) when a single grapheme must be aligned with two or more phonetic symbols. Likewise, when the NULL symbol appears in the phoneme string this correspond to a grapheme deletion. As before, we define a grapheme cluster (N+E → n) when one or more graphemes are aligned with a single phoneme.

In the previous example, if we consider a grapheme window of 4 symbols (1 left + 1 focus + 2 right), we can create the following examples for the training data base of CART, where the grapheme "~" stays for EMPTY context:

~	**H**	U	M	,	h
H	**U**	M	A	,	j+u:
U	**M**	A	N	,	m
M	**A**	N	E	,	`eI
A	**N**	E	~	,	n
N	**E**	~	~	,	-

Lexicon alignment and CART training, the two main steps involved in the training scheme of the G2P-Cart, are described in the next paragraphs.

3.1. Lexicon alignment

Lexicon alignment is the most important and critical step of the whole training scheme of the G2P-Cart system, as it builds up the data on which the CART extracts the transcription rules. The core of the process is based on the dynamic programming (DP) algorithm. DP aligns two strings finding the best alignment with respect to a distance metric between the two strings.

The lexicon alignment process iterates the application of the DP algorithm on the grapheme and phoneme sequences, where the distance metric is given by the probability that the grapheme **g** will be transcribed as the phoneme **f**, that is **P(f|g)**. The best alignment is the one with the maximum probability, that is:

$$BestPath = \max_{k}\left(\prod_{i,j\in Path_k} P(f_i \mid g_j)\right)$$

where: $Path_k$ is a generic alignment between grapheme and phoneme sequences.

The probabilities P(f|g) are estimated during training at each iteration step. In other words, starting from the best alignment found at time i, we can evaluate the new probability values and use them at time i+1. At the first iteration, the probabilities are initialized with a constant value, that is each association grapheme-phoneme has the same probability.

3.2. CART

After the alignment of the lexicon, we are able to create the training database for CART, i.e. a set of patterns where each pattern made of a number of features (numerical or categorical) and a target value. In our case, the input features describe (1) the grapheme window, (2) the position of the vowel focus with respect to the beginning/end of the word and (3) the distance from the previous/next vowel. The target is the phoneme aligned with the window focus. Starting from such data, the training algorithm of CART builds up the root node of the tree and then, selecting the best features that discriminate the output phonemes, creates the best splitting rule associated to the root. This procedure continues on the children till the training patterns associated with the current node all share the same target value. The node is then labeled as a leaf and associated with the target phoneme. If this end condition is never reached, node splitting is stopped on the basis of node cardinality and impurity and the most probable phoneme is associated with the leaf.

The whole process results in a binary tree in which the leaves are associated with output phonemes. Each path linking the root to the leaf, represents a rule in which the number of tests correspond to the number of nodes involved in the path.

This training methodology has some positive aspects. The first one is that the training procedure is able to make feature selection. This means that the most useful features to solve the problem are found in an automatic way and are used first (close to the root) in the tree. The second one is that we can use both categorical and numerical features. This means that we can mix into a single rule different kinds of information.

From an experimental point of view, the training process parameters to be tuned are the choice of the input features and the stopping criteria.

4. Experimental Results

The above described learning scheme has been extensively tested to evaluate the robustness of the technique with respect to (1) applicability to different languages, and (2) transcription accuracy, which should be adequate to the requirements of TTS and ASR systems.

We investigated four languages: British English, American English, French, Brazilian Portuguese.

For each of them a training lexical database was created starting from several sources, and merging them together. The main sources were the following: off-the-shelf electronic lexicons, research project derived lexicons (used in the field of speech recognition and synthesis), standard dictionaries and hand written phonetic transcriptions. The idea was to cover all the relevant words of each language, including not only the headwords, but also their inflected forms. Text corpora were collected, containing samples of newspapers and contemporary literature and these text data were used to perform a statistical assessment of the coverage provided by our lexicons. The most frequent words of a given language, when missing in the lexicon, were transcribed by hand by phonetic experts and added to the lexicon.

We did not use all the words of the lexicons as input to the CART algorithm, but only the subset of words which presented a regular transcription; so we had to identify, inside the lexicons, items like foreign words, abbreviations, acronyms, places, surnames, and so on. All these items were put into an exception list, and had no influence on the learning algorithm. As for the words with more than one pronunciation, their most frequent form was selected and the others were discarded. The solution of ambiguities due to non-homophone homographs was delegated to contextual rules in syntactic analysis modules.

Table I reports the summary of our lexical databases in terms of number of records. For each language the whole lexicon was split into 3 sets: the first contains all the available words (Trn100 or Tst100), while the others two contains the 80% (Trn80) and the remaining 20% (Tst20) of the whole lexicon, respectively. The Trn100 data base is used to extract the final transcription rules and to evaluate their coverage. The Trn80 is used to evaluate the generalization capability of the method: rules are extracted using 80% of the data and tested on the remaining 20% (Tst20).

Languages show different levels of complexity with respect to G2P, depending on how closely the graphemic representation of words mirrors their actual pronunciation. The mirroring is relatively close e.g. for Italian or Spanish, while in other languages (e.g. French) grapheme and phoneme transcriptions may follow different patterns which nevertheless can often be easily mapped one into the other. In other cases (e.g English) very little regularity can be found in the mapping, the phonetic transcription being strictly lexicon-dependent.

In our experiments we chose languages that differ in complexity and features.

Table I: The lexicon databases

Language	Trn100	Trn80	Tst20
British	65128	51956	13171
American	116922	93324	23597
French	38290	30558	7731
Brazilian	11236	8989	2247

4.1. British

In our G2P-Cart application to British English we faced two tasks into a single step. As lexical stress position has a strong impact on English phonetic transcription, it should be taken into account in the learning procedure. In our experiments we chose a global approach, asking the system to learn stress location and phonetic transcription altogether. Consequently, no stress mark was provided with the graphemic input. In order to provide information useful for stress prediction, we added the position of vowels in the word as an input feature. In more detail we used a grapheme window of 3+1+4, 4 vowel position features and 2 previous phoneme predictions.

4.2. American

Likewise, the American English G2P task is twofold, predicting both phonemes and stress. In this case we used a grapheme window of 4+1+5, 4 vowel position features and 2 previous phoneme predictions.

4.3. French

For French the G2P task is easier than in the case of English, at least in two senses. Phoneme-grapheme mapping is more regular. Besides, French is a fixed-accent language, where stress is not lexicon-dependent but is always carried by the last syllable in the word. So we chose to predict phonetic transcription with no stress and add stress afterwards. The input features are a grapheme window of 3+1+4, 4 vowel position, and 2 previous phoneme predictions.

4.4. Brazilian

Brazilian Portuguese is relatively a simple language with respect to phonetic transcription. Stress location is not fixed, but in our experiment we didn't predict it, as it was not present in our reference lexicon. We used a grapheme window of 3+1+4, 4 vowel position features, and 2 previous phonemes.

Table II and III report coverage and generalization performances of our G2P-Cart in the different experiments. Performances are computed at word level and a word is correctly transcribed when all phonemes involved are correct.

Table II: G2P-Cart coverage performances

Language	Experiment	Trn100
British	3+1+4, VowPos, 2PhoBack	94.79%
American	4+1+5, VowPos, 2PhoBack	96.05%
French	3+1+4, VowPos, 2PhoBack	98.38%
Brazilian	3+1+4, VowPos, 2PhoBack	99.16%

Table III: G2P-Cart generalization performances

Language	Experiment	Tst20
British	3+1+4, VowPos, 2PhoBack	66.43%
American	4+1+5, VowPos, 2PhoBack	54.56%
French	3+1+4, VowPos, 2PhoBack	85.12%
Brazilian	3+1+4, VowPos, 2PhoBack	89.50%

5. Discussion

We analyzed in detail the trees representing the transcription rules. Each path in the tree amounts to imposing constraints over the grapheme window, starting from the focus and moving one grapheme right and one left when required, and imposing constraints on the focus type and its position in the word. In other words, the transcription tree is a collection of the minimal grapheme windows necessary to correctly map the focus onto the correct phoneme in the given context.

A second consideration concerns our first choice to predict both stress and phonetic transcription in a single step. In table IV, we report more detail about the performance of the G2P-Cart system for British English on the generalization test. As you can see, a lot of mistakes are made in accent prediction: wrong position and even words with zero or two accents. This suggest that the two task may better be approached separately. We are envisaging a new set of experiments where stress position would be predicted first (by CART or by other machine learning techniques [11]) and marked on the grapheme input. The G2P task would then be easier, taking advantage of the relation between the presence of stress and the choice of the surrounding phonemes.

Finally, some comments on our error measure. As shown in Table IV, most wrong word transcriptions show a single wrong phoneme in the word (note that the mean word length in the English testing database is about 8.3 graphemes per word). The error measure reported in Tables II and III refers to the number of wrong words. This is a suitable measure when we want to know how many words would still have to be explicitly listed as exceptions in our TTS G2P application. But it doesn't tell us how severe the error is. In some cases, the right phoneme is indeed replaced by a wrong but similar phoneme. To evaluate this aspect of the performance, we used the speech recognition system as a test environment for our G2P-Cart.

Table IV: G2P-Cart British English performances

British English	**Performances**
Correct Transcription	66.43%
Wrong Transcription	33.57%
1 Phoneme Wrong	17.28%
2 Phoneme Wrong	10.14%
3 Phoneme Wrong	4.0%
Correct Accent	75.88%
Deleted Accent	1.4%
Wrong Accent	22.7%
0 Accent	6.68%
1 Accent	86.91%
2 Accent	6.2%
3 Accent	0.2%
Total Transcription	13171 words

We estimated the recognition error rate of our British English ASR system with a vocabulary of 300 words (names of UK railway stations) both with reference correct transcriptions and with CART-generated transcriptions. The reference error rate, estimated on correct vocabulary transcriptions, gives a measure of the quality of the acoustical models used in recognition. The G2P-Cart error rate was obtained relying on the G2P-Cart transcription of the recognition vocabulary. The gap between the two recognition error rates represents the acoustical distance between the G2P-Cart transcriptions and the correct ones, providing an indirect measure of how plausible they are.

Table V: G2P-Cart generalization performances

Language	**Vocabulary Size**	**Reference Error Rate**	**G2P-Cart Error Rate**
British	300	96.94%	95.99%

6. Conclusions

In this paper, we have brought some experimental evidence into the discussion concerning the application of machine learning techniques to grapheme-to-phoneme conversion. A simple CART-based approach has been implemented and experimented on several languages. The first obtained results are encouraging, although they may suggest some improvements in the learning design, and confirm that machine learning can be profitably applied to the basic G2P task in the development of multilingual speech synthesis and recognition systems.

7. References

[1] Horst-Udo H., ``*A Hybrid approach for grapheme-to-phoneme conversion based on a combination of partial string matching and a neural network*", *Proceedings of ICSLP*, Beying, 2000.

[2] Pearson S., Kuhn R., Fincke S., Kibre N., "*Automatic Methods for Lexical Stress Assignment and Syllabification", Proceedings of ICSLP*, Beying, 2000.

[3] Jensen K. J., Riis S., *"Self-Organizing Letter Code-Book for Text-To-Phoneme Neural Network Model", Proceedings of ICSLP*, Beying, 2000.

[4] Sproat R., "*Corpus-Base Methods and Hand-Built Methods", Proceedings of ICSLP*, Beying, 2000.

[5] W.Daelemans, G.Durieux, "*Inductive Lexica*", in F.VanEynde, D. Gibbon, eds., *Lexicon Development for Speech and Language Processing*, Kluwer Academic Publishers, Dordrecht, The Netherlands, 2000

[6] M. Balestri, A. Pacchiotti, S. Quazza, P.L. Salza, S. Sandri, "Choose the Best to Modify the Least: a New Generation Concatenative Synthesis System". *Proc. EUROSPEECH '99*, Budapest, 1999

[7] Fissore L., Ravera F., Laface P., "*Acoustic-phonetic Modeling for Flexible Vocabulary Speech Recognition", Proceedings of EUROSPEECH*, Madrid, 1995.

[8] Gemello R., Albesano D., Mana F., "*CSELT Hybrid HMM/Neural Networks technologies for Continuous Speech Recognition", Proceedings of IJCNN*, Como, 2000.

[9] Mana F., Quazza S., "*Text-To-Speech Oriented Automatic Learning of Italian Prosody", Proceedings of EUROSPEECH*, Madrid, 1995.

[10] P.L. Salza, "Phonetic transcription rules for text-to-speech synthesis of Italian" , Phonetica, 47, pp.66-83, 1990

[11] M. Balestri, "A coded dictionary for stress assignment rules in Italian". *Proc. EUROSPEECH '91*, Genova, 1991