Grapheme-to-Phoneme Conversion using Conditional Random Fields

Irina Illina, Dominique Fohr, Denis Jouvet

Speech Group, INRIA-LORIA, 615, rue du Jardin Botanique, 54602 Villers les Nancy, France

illina@loria.fr fohr@loria.fr jouvet@inria.fr

Abstract

We propose an approach to grapheme-to-phoneme conversion based on a probabilistic method: Conditional Random Fields (CRF). CRF give a long term prediction, and assume a relaxed state independence condition. Moreover, we propose an algorithm to the one-to-one letter to phoneme alignment needed for CRF training. This alignment is based on discrete HMMs. The proposed system is validated on two pronunciation dictionaries. Different CRF features are studied: POS-tag, context size, unigram versus bigram. Our approach compares favorably with the performance of the state-of-the-art Joint-Multigram Models for the quality of the pronunciations, but provides better recall and precision measures for multiple pronunciation variants generation.

Index Terms: speech recognition, grapheme-to-phoneme conversion, CRF models, discrete HMM.

1. Introduction

The task of grapheme-to-phoneme conversion (G2P conversion) is to automatically determine the pronunciation of a word from its written form. The main applications are automatic speech recognition, speech synthesis and other human language applications using pronunciation dictionaries. In these applications, using manually designed and verified dictionaries is the best solution. However, this is not always possible: absence of many proper names in the dictionary, sometimes their very particular pronunciation influenced by ethnic backgrounds, neologisms, and the expensive and time consuming process of manual transcription. For example, for French, we did not found a proper name pronunciation dictionary.

The difficulty of the G2P conversion task is the lack of a direct match between letters and phonemes, grapheme context dependence, the phenomenon of *liaison* in some languages (such as French) and multiple pronunciations. Thus, a “good” G2P conversion approach should take into account all of these cases and generate all possible pronunciations for each word.

Several attempts have been made to G2P conversion. Initial research was focused on rule based systems. Typically, the rules for grapheme pronunciations as a function of the context of the current letter are produced manually with the help of phoneticians. Main data-driven approaches are based on Neural Networks (NN) and decision trees. According to [1], NN and decision tree approaches are limited because they perform a local decision for each phoneme and so are clearly not optimal. Some sophisticated extensions of these approaches taking into account several preceding and following phonemes are proposed in [2] for NN and in [3] for the decision tree approach. Recent research has shown that probabilistic approaches can give consistent results: for example, using Hidden Markov Models (HMM) [4] or Joint-Multigram Models (JMM) [1]. Fully automatic approaches of HMMs [4] model the phonemes as the hidden states and graphemes as the observations. The transitions between phonemes represent the probability that one phoneme will follow another. JMMs [1] model the phonemes and graphemes together via their joint probabilities.

In our article we focus on a probabilistic approach for G2P prediction. It is based on *Conditional Random Fields* (CRF) [6]. CRFs are undirected graphical models in which each vertex represents a random variable whose distribution is to be inferred. The advantages of the CRF are a relaxed independence condition compared to HMMs, a global inference algorithm, and discriminative training. CRFs find applications in labeling and parsing of sequential data, image segmentation and as a general approach to combine features from different sources.

Recently, CRFs have been used for G2P conversion [5]. Our work differs in a number of ways:

- To train the matches between graphemes and phonemes, we use an alignment based on discrete HMMs, while JMMs are used by [5];
- We use POS (*Part-of-speech)* tagging in the CRF features;
- Our experimental investigation is larger and performed on two sets of data of different size, in two languages;
- We studied multiple pronunciations per word.

The main contributions of this article are as follows:

- Design an accurate approach based on discrete HMMs and CRFs for G2P conversion;
- Investigate the different aspects of the proposed approach on several test sets: using two languages (French and English), evaluating several feature functions, analyzing HMM-based and manual alignments;
- Study multiple pronunciations in terms of recall and precision;
- Our final results show that our system is comparable to a state-of-the-art system on a large pronunciation dictionary.

We will introduce the theoretical foundations of our G2P approach in section 2. Sections 3 and 4 present experimental results and their analysis with respect to implementation aspects on two corpora and the comparison with a state-of-the-art approach. Final discussion and conclusions are given in section 5.

2. Methodology

In our work, we propose to use CRFs for G2P prediction. CRFs [6] are a probabilistic framework for labeling and segmenting structured data, such as sequences, trees and lattices. As said before, CRFs give a long term prediction, assume a relaxed state independence condition compared to HMMs, allow discriminative training and converge to global optimum.

Our choice of CRFs is motivated by the fact that the training process will find optimal coefficients for features even if the features are correlated. Moreover, CRFs are relatively insensitive to unbalanced training and test data.

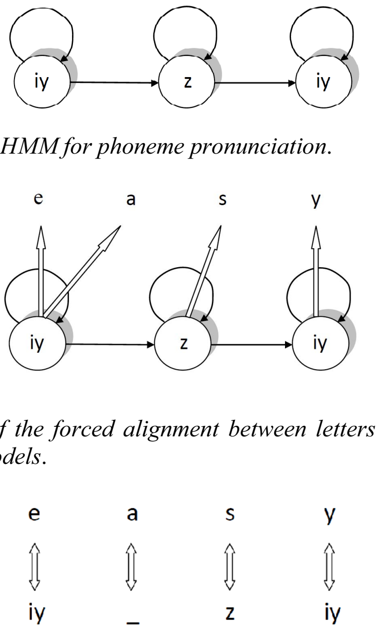
In the context of CRFs, G2P prediction training consists of two steps: the pre-processing step consists in forced aligning all words of the training dictionary in terms of grapheme-to-phoneme associations. During the second step, using the aligned training data set, G2P models are trained. Finally, these models are evaluated on a test data set. In the following, each step of the training will be described.

2.1. Pre-processing step

Grapheme-to-phoneme alignment consists in extracting the matches between one grapheme and one phoneme (or null phoneme). As CRFs need one-to-one associations (one letter to one phoneme), we proceeded in two stages: the generation of one-to-many associations, then the extraction of one-to-one matches. Figure 1 shows an example of the two stages of G2P forced-alignment for the English word "easy", whose pronunciation is /iy z iy/.

The first stage: *generation of grapheme-to-phoneme associations*. To perform the forced alignment, we propose to use discrete HMMs. We chose discrete HMMs because they allow to easily take into account the one-to-many associations (one phoneme to one or many letters). For example, for the word "easy" /iy z iy/ the first phoneme /iy/ is associated with the two first letters, "e" and "a". In our work, each phoneme is modeled by a one-state discrete HMM, each observation of this HMM corresponds to graphemes. The training of these HMMs is performed using an embedded Baum-Welch algorithm and the training part of the corpus. After training, forced-alignment between graphemes and phonemes of all words of the training corpus is performed.

The second stage: *generation of one letter to one phoneme associations*. From the previously obtained alignment, we extract the associations between one letter and one phoneme, as required by CRFs. In the case where one phoneme is aligned with several letters, this phoneme is associated with the letter with the highest probability. The remaining letters are associated with the null phoneme '-'.



(a) *Discrete HMM for phoneme pronunciation.*

(b) *Result of the forced alignment between letters and HMM phoneme models.*

(c) *Obtained one-to-one associations between letters and phonemes.*

Figure 1. *Example of letter-to-phoneme alignment for the English word "easy" /iy z iy/.*

2.2. CRF model training

The underlying idea of CRFs is defining a conditional probability distribution over label sequences given a particular observation sequence, rather than a joint distribution over both label and observation sequences.

Knowing the training letter-to-phoneme associations and some predefined feature set, CRFs learn a set of weights *w.* Learning the parameter set w is usually done by maximum likelihood learning for $p(\bar{y}|\bar{x};w)$:

$$p(\bar{y}|\bar{x};w) = \frac{1}{Z(\bar{x},w)} \exp \sum_j w_j F_j(\bar{x},\bar{y}) \quad (1)$$

$$F_j(\bar{x},\bar{y}) = \sum_{i=1}^{n} f_j(\bar{y}_{i-1},\bar{y}_i,\bar{x},i) \quad (2)$$

where $\bar{x}$ is the sequence of letters, $\bar{y}$ is the sequence of phonemes, w is the weights. f_j is a feature function and can depend on the sequence of word letters, the current phoneme, the previous phoneme, and the current position in the word. We note that unigram features correspond to $f_j(\bar{y}_i,\bar{x},i)$ in equation (2) and bigram features are represented by $f_j(\bar{y}_{i-1},\bar{y}_i,\bar{x},i)$. That is to say, unigram features take into account only the current phoneme while bigram features use the current and the previous phoneme.

During the G2P conversion, the CRF decoding algorithm can find the *n*-best sequences of phonemes corresponding to a test word.

3. Experiments

3.1. Performance metrics

Accuracy of G2P conversion is presented in term of *Phoneme error rate* (PER) and *Word error rate (WER)*. PER is computed as in [1]: the edit distance (Levenshtein distance) between the resulting and reference pronunciations is divided by the number of phonemes in the reference. WER is computed as the percentage of words that have at least one phoneme error. In the case of several reference word pronunciations, all possible pronunciations are examined and the best match is used.

3.2. Corpora

To measure the performance of the approach proposed in this paper, two pronunciation corpora were studied. The corpora are differentiated by languages (English and French), their size and number of pronunciations per word. The advantage of the NETtalk corpus is the presence of manual G2P alignment, for BDLex it is the tag information, large corpus size and multiple pronunciations for some words.

3.2.1. NETtalk English dictionary

The *NETtalk English dictionary* [7], [8] contains 19,802 words (20,008 pronunciations at all). The corpus is manually aligned in terms of letter-to-phoneme. The manual G2P alignment of NETtalk allows us to study the impact of the automatic alignment proposed in this paper using HMMs.

The phone set has 50 phonemes (including 5 double phonemes to insure a one-to-one match between phonemes and graphemes) and a null phoneme. We do not use the stress markers or syllabic markers.

To evaluate our methodology, NETtalk is partitioned randomly into disjoint training (15,000 words) and testing (5,008 words) sets. It should be noted, however, that due to the relatively small size of the NETtalk dictionary, the development set is not created. The confidence interval for 5% tolerance is ±1.3% for WER and ±0.4% for PER.

3.2.2. *BDLex French dictionary*

The BDLex French dictionary is a lexical database developed at IRIT, Paul Sabatier University, Toulouse, France [10]. It covers lexical, phonological, and morphological information. BDLex consists of about 440,000 inflected forms (generated from about 50,000 canonical words) with the following attributes: spelling, pronunciation, morphosyntactic features (part of speech, agreements, etc), canonical word spelling and a frequency indicator. The phone set consists of 38 phonemes, the letter set has 40 letters.

We divided this corpus randomly into disjoint training (75%), development (5%), and test (20%) sets. This corresponds to 263 704 words in the training set, 17581 words in the development set, and 70,322 words in the test set. The confidence interval for 5% tolerance is ±0.1% for WER and ±0.02% for PER.

3.3. Software

3.3.1. *CRF++ software*

CRF++[1] is a customizable and open source CRFs implementation for segmenting and labeling sequential data. It is written in C++, uses fast training based on gradient descent and gives *n*-best candidates.

3.3.2. *Sequitur G2P software (JMM)*

To compare our approach to state-of-the-art approaches, we chose to use the JMM approach [1]. For this, the Sequitur G2P software[2] was used. The JMM principle consists in determining the optimal set of joint sequences, where each sequence is in fact composed of a sequence of graphemes and its associated sequence of phonemes. A language model is applied on the joint sequences. The algorithm proceeds in an incremental way. The initial pass creates a very simple model. Then, each pass relies on the previously created model to enlarge the joint sequences whenever relevant to do so. In the reported experiments, 6 passes were applied, and the model having the best results on the development set was used for obtaining the pronunciations of the lexicon entries.

4. Experimental results

4.1. NETtalk dictionary results

The goal of these experiments is to evaluate the CRF approach using two kinds of alignment: the manual letter-to-phoneme one (labeled "Manual alignment" in Table 1) provided by the NETtalk corpus and the alignment obtained using discrete HMMs ("HMM alignment").

Table 1 contains the results for the NETtalk test using the best configuration obtained on the BDLex corpus (see next subsection). Comparing these results, we observe a non significant slight performance decrease for discrete HMM alignment (from 35.0% to 35.7% WER). The state-of-the-art JMM approach gives 34% WER (8.5% PER)[3]. This result is very close to the CRF results. This confirms the efficiency of our approach to perform G2P conversion.

Table 1. *CRF G2P accuracy for NETtalk test set*

System	PER(%)	WER(%)
Manual alignment	**8.2**	**35.0**
HMM alignment	8.4	35.7

4.2. BDLex French dictionary results

The large size of the BDLex dictionary, POS-tag information, and careful checking of this dictionary allow us to assess our G2P conversion system. We use automatic letter-to-phoneme alignment with HMMs because we have no available manual alignment for BDLex.

In order to find a "good" parameter set, the development set of the corpus is explored. The obtained parameters are then applied on the test set. Table 2 presented below depict the results on the test set using these parameters.

4.2.1. *Influence of POS-tag and context*

First, we investigate the influence of POS-tags on G2P conversion performance. We fixed the grapheme context to three, i.e. previous letter, current letter and following letter (labeled "±1" in Table 2). Our two POS-tags correspond to "verb" for verbs and "non verb" for other words, because in French, the written forms of some finite verb forms can be the same as the written forms of some names, but at the same time, their pronunciations are different. In Table 2 we observe that adding the POS-tag feature reduces WER from 45.6 to 41.2%. We decided to include this tag in the remaining experiments. Table 2 also shows that the larger the grapheme context, the better the results are. For example, from 41.2% WER for ±1 context (line 2 of Table) we get 7.6% for ±4 context (line 5 of Table).

4.2.2. *Effect of unigram and bigram features*

Next, we study the effect of unigram and bigram features. Unigram features take into account only the current phoneme while bigram features use the current and previous phonemes. Results from Table 2 (labeled "unig" and "bigr") suggest that it is much better to use bigram features that unigram ones: 1.2% WER versus 7.6% for ±4 context represents a statistically significant improvement. However, with bigrams, the number of features is strongly increased (see column 2), which leads to considerable memory consumption.

Now, we would like to show how a very large feature set can influence the G2P quality. For this, we have assembled unigram and bigram features and different sizes of grapheme contexts in the same CRF model ("Tag, unig, ±1±2±3±4, bigr, ±1±2±3±4" in Table 2). This gives our best result: 1.0% WER. This result is somewhat better than the previous one obtained with "Tag,bigr±4" configuration. The best configuration will be used for the remaining experiments.

Two last lines of Table 2 present the influence of the training set size on the results: the best configuration is trained on 36,000 words and on 130,000 instead of 263,704 (labeled as "train 36,000" and "train 130,000). For "train 36,000", we observe a 2.9% WER (0.5% PER) and conclude that when using only one tenth of the training set, the performance of the G2P system is not so bad and even better than any unigram configuration.

[1] crfpp.sourceforge.net
[2] www-i6.informatik.rwth-aachen.de/web/Software/g2p.html
[3] We cannot use the JMM results presented in [1] on NETtalk corpus because our train/test partition is different. So we ran JMM on our data.

To compare the performance of the proposed G2P conversion approach to state-of-the-art results, we performed G2P conversion on the same data using the Bisani-Ney JMM approach [1], [11], [12] using the Sequitur software (see section 3.3.2). The performances obtained by JMM and our approach are very similar: 0.9% WER (0.2% PER), and 1.0% WER (0.2% PER) respectively. From these results we can conclude that on the BDLex dictionary our approach obtained state-of-the-art approach performance.

We note that the WER on the English dictionary is far more important that the French BDLex corpus results. We attribute this to the fact that French word pronunciation is more regular and more predictable that in English.

Table 2. *CRF G2P accuracy for BDLex test set*

Model	Nbr Features	PER (%)	WER (%)
No tag, unig, ±1	5490	8.0	45.6
Tag, unig, ±1	10710	7.1	41.2
Tag, unig, ±2	18090	1.8	11.6
Tag, unig, ±3	25605	1.4	8.8
Tag, unig, ±4	33300	1.2	7.6
Tag, bigr, ±1	6283575	1.4	9.4
Tag, bigr, ±2	14035275	0.5	3.3
Tag, bigr, ±3	22493700	0.3	1.5
Tag, bigr±4	31182975	0.2	1.2
Tag, unig, ±1±2±3±4, bigr, ±1±2±3±4	51414975	**0.2**	**1.0**
Tag, unig, ±1±2±3±4, bigr, ±1±2±3±4, train 130000	51414975	0.3	1.4
Tag, unig, ±1±2±3±4, bigr, ±1±2±3±4, train 36000	51414975	0.5	2.9

4.2.3. *Multiple pronunciation variants generation*

As said before, a "good" G2P conversion approach should generate all possible pronunciation variants for every word. In the previous experiments, only one pronunciation per word was generated. In this section, we generate multiple pronunciations per word and study their quality compared to the reference multiple pronunciations in the corpus. By varying a decision threshold, we generate one or several pronunciation variants per word based on their probabilities.

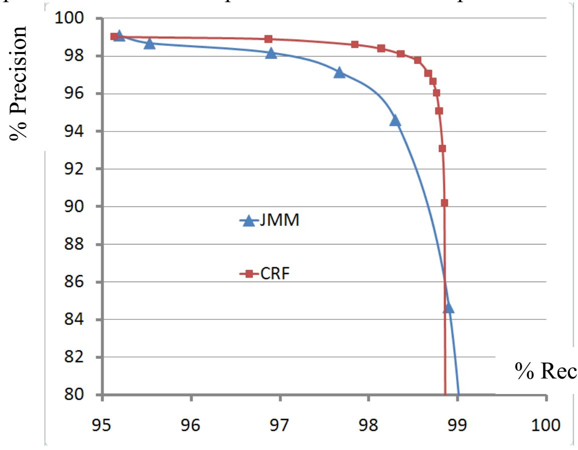


Figure 2. *Recall and precision for JMM and CRF G2P conversion using multiple pronunciations, BDLex test set.*

The performance metric used is based on the recall precision measures. The recall (X axis in Figure 2) is the number of correct pronunciation variants generated divided by the total number of reference pronunciation variants; precision (Y axis) is the number of correct pronunciation variants divided by the total number of generated pronunciation variants. Figure 2 presents the recall and precision for different threshold values and shows that the generated pronunciations are relevant: the precision stays very good (>98%) and recall increases (up to 98.4 %). Moreover, CRF performance is better that of JMM.

5. Conclusions

In this paper we proposed an algorithm based on CRFs that realizes an efficient grapheme-to-phoneme conversion. This approach needs a one-to-one letter-to-phoneme alignment. This one-to-one matching is provided by a discrete HMM. Experiments with the NETtalk dictionary, where manual alignment is provided, show statistically similar performance using manual or HMM-based alignment. This validates our automatic alignment approach, which is necessary for dealing with large dictionaries.

To assess our CRF-based G2P system, experiments on two small and large dictionary corpora and on two languages were conducted. Different CRF parameters were studied: POS-tag, context size, unigram versus bigram. The best CRF configuration was identified and compared to a state-of-the-art JMM system. Our approach compares favorably with the performance of the state-of-the art JMM technique. Concerning multiple pronunciation variant generation, our system demonstrated a superior precision and recall performance to JMM and so allows to better model the multiple variants.

6. References

[1] Bisani, M., Ney, H., "Joint-Sequence Models for Grapheme-to-Phoneme Conversion", Speech Communication, 50: 434-451, Elsevier, 2008.

[2] Jensen, K.J., Riis, S., "Self-Organizing Letter Code-Book for Text-to-Phoneme Neural Network Model", Proc. International Conference on Spoken Language Processing, 3, 318-321,2000.

[3] Pagel, V., Lenzo, K., Black, A.W., "Letter-to-Sound Rules for Accented Lexicon Compression", Proc. International Conference on Spoken Language Processing, 5, 2015-2018, 1998.

[4] Taylor, P. "Hidden Markov Models for Grapheme to Phoneme conversion", Proc. Interspeech, 1973-1976, 2005.

[5] Wang, D., King, S., "Letter-to-sound Pronunciation Prediction Using Conditional Random Fields", IEEE Signal Processing Letters, 18(2):122-125, 2011.

[6] Lafferty, J., McCallum, A., Pereira, F. "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data", Proc. International Conference on Machine Learning, 282-289, 2001.

[7] Sejnowski, T.J., Rosenberg, C.R., "NETtalk corpus. " ftp://svr.ftp.eng.cam.ac.uk/pub/copm.speech/dictionaries.

[8] Sejnowski, T.J., "The NETtalk Corpus: Phonetic Transcription of 20,008 English Words", 1988.

[9] Weide, R.L., "The Carnegie Mellon pronunciation dictionary". http://www.speech.cs.cmu.edu/cgi-bin/cmudict.

[10] De Calmès, M., Perennou, *G.* "BDLEX : a Lexicon for Spoken and Written French." 1st International Conference on Language Resources & Evaluation (LREC), Granada, 1129-1136, 1998.

[11] Galescu, L., Allen, J.F., "Pronunciation of Proper Names with a Joint N-Gram Model for Bi-Directional Grapheme-to-Phoneme Conversion", Proc. International Conference on Spoken Language Processing, vol.1, 109-112, 2002.

[12] Galescu, L., Allen, J.F., "Bi-Directional Conversion Between Graphemes and Phonemes Using a Joint N-Gram Model", Proc. 4th ISCA Tutorial and Research Workshop on Speech Synthesis, Perthshire, 2001.