INCORPORATING ALIGNMENTS INTO CONDITIONAL RANDOM FIELDS FOR GRAPHEME TO PHONEME CONVERSION

Patrick Lehnen *Stefan Hahn* *Andreas Guta* *Hermann Ney*

{lehnen,hahn,guta,ney}@cs.rwth-aachen.de
Human Language Technology and Pattern Recognition
Computer Science Department, RWTH Aachen University, 52056 Aachen, Germany

ABSTRACT

Conditional Random Fields (CRFs) are a state-of-the-art approach to natural language processing tasks like grapheme-to-phoneme (g2p) conversion which is used to produce pronunciations or pronunciation variants for almost all ASR pronunciation lexica. One drawback of CRFs is that for training, an alignment is needed between graphemes and phonemes, usually even 1-to-1. The quality of the g2p result heavily depends on this alignment. Since these alignments are usually not annotated within the corpora, external models have to be used to produce such an alignment in a preprocessing step. In this work, we propose two approaches to integrate the alignment generation directly and efficiently into the CRF training process. Whereas the first approach relies on linear segmentation as starting point, the second approach considers all possible alignments given certain constraints. Both methods have been evaluated on two English g2p tasks, namely NETtalk and Celex, on which state-of-the-art results have been reported in the literature. The proposed approaches lead to results comparable to the state-of-the art.

Index Terms— CRF, G2P, Alignments, EM Algorithm

1. INTRODUCTION

Conditional Random Fields (CRFs) represent a powerful, discriminative modelling framework, leading to state-of-the-art results for natural language processing tasks. They can easily be applied to all monotone string-to-string translation tasks, where a 1-to-1 alignment between source and target side is available. Grapheme-to-phoneme conversion (g2p) represents such a task. Here, for a given sequence of graphemes, a phoneme sequence representing a valid pronunciation has to be generated. This is an important task for almost all state-of-the-art ASR systems, since especially the pronunciations of named entities are usually not given in a standard lexicon. One constraint for the use of CRFs is that a 1-to-1 alignment between graphemes and phonemes is necessary to train the model. The alignment is usually provided by an external model and can easily be transferred to a 1-to-1 alignment.

The need for such an alignment results in two drawbacks: first, an additional model has to be trained and tuned and second, there may be a mismatch between the alignment provided by an external model and an alignment suited for CRF training. It would be desirable to get rid of the external alignment and the possible error propagation as well as the additional tuning steps.

The work in this publication was inspired by current success of log-linear modelling for g2p conversion shown by [1], the success of CRFs in a wide range of applications, e.g. concept tagging [2], g2p [3], and the earlier work on Hidden Conditional Random Fields [4, 5] (HCRFs) and Hidden Dynamic Conditional Random Fields [6] (HDCRFs). HCRFs as described by [4] and [5] sum over a predefined graph capturing the hidden structure. In [4] the graph is a mesh between features, while [5] uses a part of a speech parse tree. HDCRFs as described in [6] are similar to our approach specified in Sec. 3.2. They included hidden variables by summing up over all alignments in training. To keep training feasible they limited the set of hidden states per seen state. However they only reported result on machine learning tasks like part-of-speech (POS) tagging and named entity recognition (NER).

In contrast to the experiments reported in the literature, we will show experiments on g2p tasks with much more output labels than e.g. POS tagging or NER, will extend the EM-Algorithm to CRFs, and will show an efficient way to implement our approach. Except the restriction to monotonic alignments with many source symbols to one target symbol, our approach does not need any external knowledge or restrictions and finds the alignment only by applying CRFs.

For comparison with external alignment models, giza++ [7] and the joint n-gram approach [8, 9] have been used to produce alignments on which a conventional CRF has been trained.

In the next section, CRFs are introduced in detail followed by a section describing the two approaches which have been realized to incorporate the alignment into CRFs. In Sec. 4, experimental results and a comparison to state-of-the art approaches are presented. The last two section of the paper give a conclusion and an outlook. The paper concludes with Sec. 5.

2. CONDITIONAL RANDOM FIELDS

Linear Chain Conditional Random Fields (CRFs) introduced by [10] are defined as the conditional probability of a target sequence $t_1^N = t_1, \ldots, t_N$ given a source sequence $s_1^N =$

This work was partly realized as part of the Quaero Programme, funded by OSEO, French State agency for innovation.

$s_1, \ldots, s_N$ using a log-linear representation:

$$p(t_1^N|s_1^N) = \frac{\exp H(t_1^N, s_1^N)}{\sum_{\tilde{t}_1^N} \exp H(\tilde{t}_1^N, s_1^N)} \tag{1}$$

$$H(t_1^N, s_1^N) = \left(\sum_{n=1}^{N} \sum_{l=1}^{L} \lambda_l h_l(t_{n-1}, t_n, s_1^N) \right) \tag{2}$$

$H(t_1^N, s_1^N)$ defines position dependent feature functions $h_l(t_{n-1}, t_n, s_1^N)$. In the experiments they were binary ($\in 0, 1$) functions, were lexical features ($t_n = t', s_{n+\epsilon} = s'$), bigram features ($t_{n-1} = t'', t_n = t'$), and any "and"-combinations of them (to capture n-grams) are possible. The training criteria over a training dataset $\{\{\bar{t}_1^N\}_k, \{s_1^N\}_k\}_{k=1}^K$ is given by the maximization of the conditional log-likelihood L

$$L = \sum_{k=1}^{K} \log p(\{\bar{t}_1^N\}_k | \{s_1^N\}_k) - c_2 ||\lambda_1^M||_2^2 \tag{3}$$

using a L2-regularization constant c_2, while the decision criteria is given by the maximization of the sentence wise probability $p(t_1^N|s_1^N)$.

In [11], the idea of merging the optimization of feature weights (training) based on SVMs and CRFs, called MMI there, is described. This is realized by modifying the potential function H in training to:

$$H \rightarrow \hat{H}(t_1^N, s_1^N) = H(\tilde{t}_1^N, s_1^N) - \rho \mathcal{A}(\tilde{t}_1^N, \bar{t}_1^N) \tag{4}$$

Here, the margin score is set to the word accuracy $\mathcal{A}(t_1^N, \bar{t}_1^N) = \sum_{n=1}^N \delta(t_n, \bar{t}_n)$ between the hypothesis t_1^N and the reference $\bar{t}_1^N$, scaled by $\rho \geq 0$.

3. ALIGNMENTS

CRFs as described in the previous section assume the same length of the source s_1^N and target sequence τ_1^M ($N = M$). Thus we start by the traditional approach of integrating a hidden alignment variable a_1^N:

$$p(\tau_1^M|s_1^N) = \sum_{a_1^M} p(\tau_1^M, a_1^M|s_1^N) \tag{5}$$

It is possible to model the tuple (τ_1^M, a_1^M) by a projection using the so-called BIO scheme, proposed in [12]. For each source symbol s_n a tag t_n is a tuple of the aligned target symbol τ_m and a "begin" (B) or "inside" (I) marker. An example of a possible 1-to-1 alignment for a word/pronunciation pair using this scheme would look like this:

"throw"	=	t	h	r	o	w
[θrɐ]		θ_B	θ_I	r_B	ɐ_B	ɐ_I

These scheme allows the modelling of monotone alignments with the restriction of one target symbols to one and to many source symbols, but not vice versa. Taking our task of grapheme to phoneme conversion this restrictions can be accepted. Using the BIO scheme permits CRFs to model the probability $p(\tau_1^M, a_1^M|s_1^N) = p(t_1^N|s_1^N)$.

There are two ways of implementing Eq. 5: First the sum can be approximated by a maximum resulting to an EM-like algorithm, described in Sec. 3.1, or second the sum can be computed directly, described in Sec. 3.2.

3.1. Maximum Approach

Starting from a linear segmentation

$$pos(s) \rightarrow (pos(s) \cdot len(t)/len(s)) \mod len(t)$$

a CRF is trained using the tag sequence t_1^N of the correct target sequence τ_1^M and the linear segmentation a_1^M as correct/reference sequence

$$p(t_1^N|s_1^N)|_{t_1^N = t_1^N(\tau_1^M, a_1^M)} = \frac{\exp H(t_1^N, s_1^N)}{\sum_{\tilde{t}_1^N} \exp H(\tilde{t}_1^N, s_1^N)}, \tag{6}$$

which is called maximization step in EM-Training. The trained model is applied on the training corpus with restricting the search to the correct target sequence τ_1^M

$$\hat{a}_1^M = \underset{a_1^M}{\operatorname{argmax}} \left\{ p(t_1^N(\tau_1^M, a_1^M)|s_1^N) \right\}, \tag{7}$$

which is called expectation step in EM-Training. Training continues by a CRF training/resegmentation loop (Eqs. 6 and 7) until convergence.

In the maximization step the convexity property of CRFs is still preserved, however by application of the expectation step the convexity is broken. In early experiments it turned out, that it was useful to keep the model used in Eq. 6 simple and blurred to avoid local optima. In the final maximization step the full CRF model using all features were trained until convergence.

3.2. Summation Approach

Combining Eqs. 1 and 5 under use of the BIO scheme tags t_n results to

$$p(\tau_1^M|s_1^N) = \frac{\sum_{a_1^M} \exp H(\tau_1^M, a_1^M, s_1^N)}{\sum_{\tilde{a}_1^M} \sum_{\tilde{\tau}_1^M} \exp H(\tilde{\tau}_1^N, \tilde{a}_1^M, s_1^N)} \tag{8}$$

$$= \frac{\sum_{t_1^N : a_1^M} \exp H(t_1^N, s_1^N)}{\sum_{\tilde{t}_1^N} \exp H(\tilde{t}_1^N, s_1^N)} \tag{9}$$

In training the conditional log-likelihood L (Eq. 3) splits up to three summands: two corresponding to the numerator and denominator of $p(\tau_1^M|s_1^N)$ and one corresponding to the regularization term. In comparison to regular linear chain CRFs only the numerator summand is changed, due to the summation over all alignments. The numerator summand gets the same structure as the denominator summand and can be solved by the same posterior approach (using Forward-/Backward Algorithm). Since the sum in the numerator of Eq. 8 is a restricted variant of the denominator sum the computational

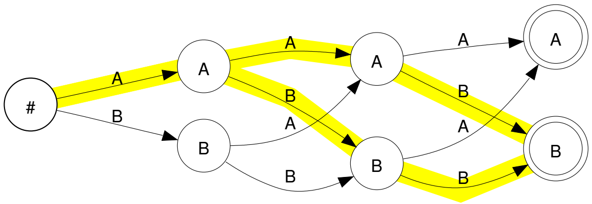

Fig. 1. Automaton describing the topology of all possible tag sequences t_1^3 with length 3 and vocabulary $\tau \in \{A, B\}$. To keep the diagram simple BIO markers are not included. The correct paths for target sequence $\tau_1^2 = [A, B]$ is marked yellow/grey.

cost is in worst case doubled, which is acceptable. Unfortunately in search we have a mixture of summation and maximisation which results to exponential behaviour making it hard to carry out the alignment sum in search. Thus we only applied the maximisation for both target and alignment sequence.

The advantage of this summation approach is that it is mostly configuration free. It is not necessary to tune an alignment before, the alignment is a true hidden variable. Its disadvantage however is that it breaks the convexity of the training criteria.

3.3. Implementation using FSAs

The methods described in Sec. 3.1 and 3.2 are modelled using finite state automata (FSA). An input sequence s_1^N encoded as linear chain FSA is augmented by the tag vocabulary (target vocabulary times BIO scheme markers) as output labels, and composed with a bigram automaton guaranteeing that all arcs pointing to one state have the same tag (output symbol). An example of a resulting automaton with $N = 3$ and $\tau \in \{A, B\}$ is sketched in Fig. 1. An FSA-posterior operation with Log-semiring is used for the denominator part of Eq. 3. The numerator part for Sec. 3.2 is realized by selecting all paths expressing the correct target sequence τ_1^M (see yellow/grey paths in Fig. 1) and again applying FSA-posterior with Log-semiring. The resegmentation part of Sec. 3.1 is an FSA-best with tropical-semiring on this selected paths (again yellow/grey paths in Fig. 1).

4. EXPERIMENTS

In this section, experiments on two publicly available English g2p corpora are reported. The statistics are given in Tab. 1. The NETtalk corpus [13] is a comparatively small one with roughly 15k grapheme/phoneme word pairs, whereof 1k has been set aside as development set for tuning. An additional advantage of this corpus is that a manual alignment is available, which is rarely the case for g2p corpora. The Celex corpus [14] has roughly 40k training words, and with 15k words a bigger test set. Thus, even small improvements of the model can be measured, since the error rates are comparatively low on this corpus. Other authors have used exactly these corpora and data splits, thus a comparison of the proposed methods with the state-of-the-art is possible.The results are presented w.r.t. phoneme error rate (PER) and word error rate (WER). For scoring, the NIST scoring toolkit has been applied [15].

In a first experiment, we wanted to investigate the effect of the alignment on the performance of the CRF model. We tested four different external models to produce alignments, which will now be shortly described. A straight-forward alignment can be realized using a linear segmentation (cf. Sec. 3.1 for the corresponding equation). A popular tool to produce alignments for machine translation is giza++. We utilized this tool to produce an alignment using the following sequence of models: 4x IBM1, 4x HMM, 2x IBM3, 2x IBM4, 3x IBM5 (see e.g. [7]). The joint n-gram approach as presented in [9] is also often used for enriching ASR lexica with pronunciation variants and automatically derived pronunciations. We use exactly the best models presented in the aforementioned paper. Since manual alignments are available for the NETtalk corpus [13], they are also considered.

For each of the resulting alignments, a CRF is trained with exactly the same features. Thus, only the influence of the alignment can be observed. The results are presented in Tab. 2. For both corpora, the linear segmentation leads to the worst results as expected. On NETtalk, giza++, the joint n-gram and the manual alignment give roughly the same result, namely a PER of around 7.5%. These figures are roughly 6% relatively better than the numbers reported in [9]. This huge improvement can be traced back to the fact that usually discriminative methods work better than generative models, if little training data is available. On Celex, there is now a significant difference in the performance of the giza++ and the joint n-gram alignment, whereas the latter can improve the giza++ result by roughly 30% relatively. Here, the results are comparable to the numbers presented in [9] and [1], whereas the latter publication gives the best result on this g2p task in the literature.

These results are now the baseline for grading of the integrated alignment approaches. On both corpora, the maximum approach as well as the summation approach have been tested. For the former, a broader model is utilized for the resegmentation step as explained in Sec. 3.1. We only used lexical features in a window of $[-1, \ldots, 1]$ around the current word and the bigram feature. We did a resegmentation after iterations 5, 10, 15, 25, 35, 45, 65, 85 and 105. Additionally, the lambdas and step-sizes have been reseted. With such a small feature set,this procedure is pretty fast, albeit the number of iterations is high. With the final alignment, a CRF model has been trained for 50 iterations using a much larger feature set, also incorporating combined features and a larger lexical window size. It is the same feature set which has been utilized and optimized for the various alignments. For the summation approach, the model is simply trained with the refined feature set for 50 iterations. These experimental results are also presented in Tab. 2. On NETtalk, one can see that the maximum approach could improve the linear segmentation to the quality of the manual alignment. The summation approach did not perform this well, but the results are better than linear segmentation. On Celex, the picture is similar. The maxi-

Table 1. Corpus statistics for the two considered Englisch g2p corpora including the partitioning of the data into training, development end evaluation sets.

	symbols		number of words		
	$\|S\|$	$\|T\|$	train	test	dev
NETtalk 15k	26	50	13804	4951	1071
Celex	26	53	39995	15000	5000

Table 2. Effect of various alignments on two g2p tasks.

data set	alignment	PER [%]		WER [%]	
		Dev	Eva	Dev	Eva
NETtalk 15k	linear	10.1	10.6	43.7	44.9
	giza++	7.6	8.0	33.9	34.5
	joint n-gram	7.4	7.9	33.2	34.2
	manual	7.6	7.8	33.6	33.7
	CRF max	7.5	7.9	34.0	34.1
	CRF sum	9.0	9.5	39.7	39.8
Celex	linear	5.3	4.9	25.1	23.6
	giza++	3.7	3.6	18.8	18.1
	joint n-gram	2.6	2.5	13.0	12.4
	CRF max	2.9	2.8	14.6	13.9
	CRF sum	3.2	3.0	15.3	14.4

mum approach is comparable to the joint n-gram approach. Here, the summation approach is on the same level as giza++, a little worse than the joint n-gram approach.

5. CONCLUSION

In this paper, we have presented two approaches to integrate the alignment into the CRF training process. The first approach uses two steps and the maximum approximation, where an initial alignment is iteratively improved. The second algorithm takes all possible alignments (approximated by certain constraints) into account. Both methods give results which are comparable to state-of-the-art results, whereas the first approach outperforms the second one, even getting the same performance as with a given manual alignment for the NETtalk corpus.

6. REFERENCES

[1] Sittichai Jiampojamarn and Grzegorz Kondrak, "Online discriminative training for grapheme-to-phoneme conversion," in *Proceedings of ISCA Interspeech*, Brighton, U.K., Sept. 2009, pp. 1303–1306.

[2] Stefan Hahn, Patrick Lehnen, Georg Heigold, and Herman n Ney, "Optimizing CRFs for SLU Tasks in Various Languages Using Modified T raining Criteria," in *Proceedings of ISCA Interspeech*, Brighton, U.K., Sept. 2009, pp. 2727–2730.

[3] Thomas Lavergne, Olivier Cappé, and François Yvon, "Practical very large scale crfs," in *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2010.

[4] Ariadna Quattoni, Sybor Wang, Louis-Philippe Morency, Michael Collins, and Trevor Darrell, "Hidden conditional random fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, pp. 1848–1852, 2007.

[5] Terry Koo and Michael Collins, "Hidden-variable models for discriminative reranking," in *HLT '05: Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, Morristown, NJ, USA, 2005, pp. 507–514, Association for Computational Linguistics.

[6] Xiaofeng Yu and Wai Lam, "Hidden dynamic probabilistic models for labeling sequence data," in *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence*, Chicago, IL, USA, July 2008, pp. 739–745.

[7] Franz Josef Och and Hermann Ney, "A systematic comparison of various statistical alignment models," *Computational Linguistics*, vol. 29, no. 1, pp. 19–51, 2003.

[8] Sabine Deligne, François Yvon, and Frédéric Bimbot, "Variable-length sequence matching for phonetic transcription using joint multigrams," in *Proceeding of the Fourth European Conference on Speech Communication and Technology (EUROSPEECH)*, Madrid, Spain, Sept. 1995, pp. 2243–2246.

[9] Maximilian Bisani and Hermann Ney, "Joint-sequence models for grapheme-to-phoneme conversion," *Speech Communication*, vol. 50, no. 5, pp. 434–451, May 2008.

[10] J. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proceedings of the Eighteenth International Conference on Machine Learning (ICML)*, Williamstown, MA, USA, June 2001, pp. 282–289.

[11] G. Heigold, R. Schlüter, and H. Ney, "Modified MPE/MMI in a Transducer-Based Framework," in *Proceedings of the Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, Taipei, Taiwan, Apr. 2009.

[12] Lance Ramshaw and Mitchell Marcus, "Text Chunking using Transformation-Based Learning," in *Proceedings of the 3rd Workshop on Very Large Corpora*, Cambridge, MA, USA, June 1995, pp. 84–94.

[13] T J Sejnowski and C S Rosenberg, "Parallel networks that learn to pronounce english text," *Complex Systems*, vol. vol. 1, pp. 145–168, Feb 1987.

[14] R.H. Baayen, R. Piepenbrock, and L. Gulikers, "Celex2," 1996.

[15] NIST, "Speech Recognition Scoring Toolkit (SCTK)," http://www.nist.gov/speech/tools/.