ISCA Archive
http://www.isca-speech.org/archive

INTERSPEECH 2012
ISCA's 13th Annual Conference
Portland, OR, USA
September 9-13, 2012

Hidden Conditional Random Fields with M-to-N Alignments for Grapheme-to-Phoneme Conversion

Patrick Lehnen, Stefan Hahn, Vlad-Andrei Guta, Hermann Ney

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

Abstract

Conditional Random Fields have been successfully applied to a number of NLP tasks like concept tagging, named entity tagging, or grapheme-to-phoneme conversion. When no alignment between source and target side is provided with the training data, it is challenging to build a CRF system with state-of-the-art performance. In this work, we present an approach incorporating an M-to-N alignment as a hidden variable within a transducer-based implementation of CRFs. Including integrated estimation of transition penalties, it was possible to train a state-of-the-art hidden CRF system in reasonable time for an English grapheme-to-phoneme conversion task without using an external model to provide the alignment.
Index Terms: CRF, G2P, Alignment, M-N

1. Introduction

Conditional Random Fields (CRFs) have been applied to many NLP-related tasks in recent years, e.g. grapheme-to-phoneme (G2P) conversion [1], concept tagging [2], or name transliteration [3]. One issue which has often been neglected is how to cope with tasks where no alignment between source and target side is provided with the training data. One example is given in Fig. 1. Here, the letter "x" has to be aligned to the two phonemes "k" and "s".

The obvious solution has often been to use an external model to produce such an alignment, e.g. with giza++ [4] or even CRFs [5], which model the probability of an alignment $a_1^M = a_1, \ldots, a_M$, given the source sequence $x_1^N = x_1, \ldots, x_N$ and the target sequence $y_1^N = y_1, \ldots, y_N$: $p(a_1^M | x_1^N, y_1^N)$. But in search the target is not known, resulting in a very different probability $p(y_1^N, a_1^M | x_1^N)$ or $p(y_1^N | x_1^N) = \sum_{a_1^M} p(y_1^N, a_1^M | x_1^N)$. In most published approaches this mismatch needs to be circumvented by heuristics. In this publication we model $p(y_1^N | x_1^N) = \sum_{a_1^M} p(y_1^N, a_1^M | x_1^N)$ in training and $p(y_1^N, a_1^M | x_1^N)$ in search.

Whereas there are already ways to extend the CRF approach to produce N-1 alignments using the so-called *BIO scheme* [6], the disadvantage of the N-1 alignments is that target sequences longer than the source sequence can not be modeled. We propose a CRF model integrating M-N alignments.

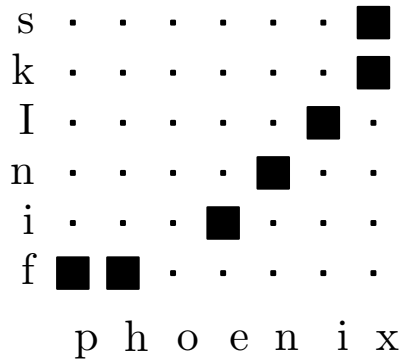


Figure 1: Example of an alignment between a grapheme and a phoneme sequence.

The proposed method is evaluated on the English Celex G2P task. G2P tasks have the advantage that a great amount of experiments and variants of HCRFs can be evaluated in reasonable time. The results are not restricted to G2P but can be applied to all similar but more complex tasks with respect to data sizes, e.g. machine translation.

In the next section, CRFs are presented followed by a brief description of our FSA-based implementation. N-1 alignments are introduced next, followed by the M-N alignment approach. The paper concludes with experimental results and their discussion.

2. Linear Chain Conditional Random Fields

Linear Chain Conditional Random Fields (CRFs) introduced in [7] are defined as the conditional probability of a target sequence $y_1^N = y_1, \ldots, y_N$ given a source sequence $x_1^N = x_1, \ldots, x_N$ using a log-linear representa-

tion:

$$p(y_1^N|x_1^N) = \frac{\prod_{n=1}^{N} e^{H(y_{n-\delta}^n, x_1^N)}}{\sum_{\tilde{y}_1^N} \prod_{n=1}^{N} e^{H(\tilde{y}_{n-\delta}^n, x_1^N)}} \quad (1)$$

$$H(y_{n-\delta}^n, x_1^N) = \sum_{l=1}^{L} \lambda_l h_l(y_{n-\delta}^{n-1}, y_n, x_1^N) \quad (2)$$

$H(y_{n-\delta}^n, x_1^N)$ defines position dependent feature functions $h_l(y_{n-\delta}^{n-1}, y_n, x_1^N)$ (details are given in Sec. 2.1). The training criterion over a training dataset $\{\{\bar{t}_1^N\}_k, \{x_1^N\}_k\}_{k=1}^K$ is given by the maximization of the conditional log-likelihood L:

$$L = \sum_{k=1}^{K} \log p(\{\bar{t}_1^N\}_k | \{x_1^N\}_k) - \sum_{i=1}^{2} c_i ||\lambda_1^M||_i^i$$

using L1- and L2-regularization constants c_1, c_2, while the decision criterion is given by the maximization of the sentence wise probability $p(y_1^N|x_1^N)$. For details about the derivation of the regularization cf. e.g. [8].

2.1. Features

In the described experiments, the feature functions $h_l(y_{n-\delta}^{n-1}, y_n, x_1^N)$ are binary features ($\in 0, 1$), having source to target features depending on source symbols and the current target symbol (y_n, x_m) with some m, "and"-combinations of these features $(y_n, x_{m-\gamma_1}^{m+\gamma_2})$, transition features $(y_{n-\delta}^{n-1}, y_n)$ of different length δ. To gain sparseness, we only select features seen at least once in the training set.

2.2. Finite State Automata Design

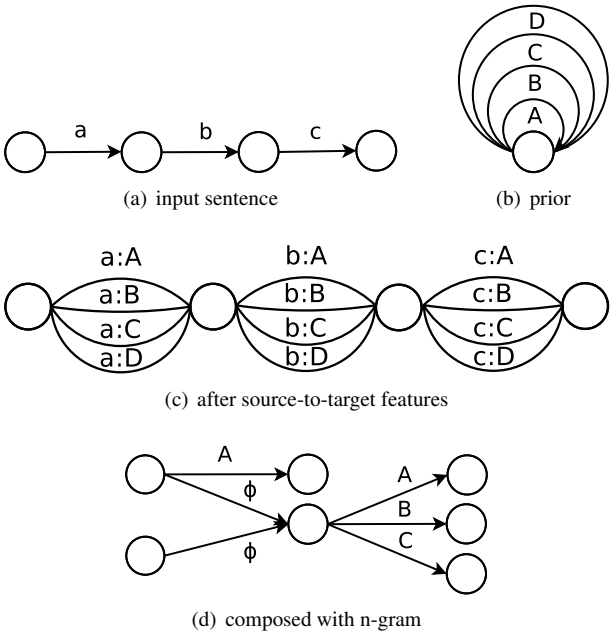
The CRFs were implemented using Finite State Automata. The source symbol sequence is represented as a chain (in Fig. 2(a) as `abc`). A prior state holds one arc per possible target symbol (in Fig. 2(b) as `A`, `B`, `C`, and `D`). The input chain is augmented by the target vocabulary and weighted by prior and source-to-target features. The resulting automaton (Fig. 2(c)) is finally composed with an n-gram automaton weighted with the target n-gram features (Fig. 2(d)). To benefit from the sparsity in the features, Φ/failure arcs (cf. [9]) are used [10, 1].

3. 1-to-N Alignments

It has already been shown that it is possible to integrate a hidden variable such as an alignment in CRFs:

$$p(y_1^N|x_1^M) = \frac{\sum_{a_1^M} \prod_{n=1}^{N} e^{H(a_m, y_{n-\delta}^n, x_1^M)}}{\sum_{\tilde{a}_1^M} \sum_{\tilde{y}_1^N} \prod_{n=1}^{N} e^{H(\tilde{a}_m, \tilde{y}_{n-\delta}^n, x_1^M)}} \quad (3)$$

Depending on the authors this integration is called Hidden Conditional Random Fields [11, 12] (HCRFs) or Hidden Dynamic Conditional Random Fields [13] (HDCRFs). In [14] it is shown that the computational cost for calculating a HCRF is not significantly changed compared to a CRF with given Alignment, as the numerator is a subset of the denominator. In the latter publication, the alignment is realized by adopting the *BIO scheme* [6], where the abbreviation stands for "begin" (B), "inside" (I), and "outside" (O) markers (cf. e.g. [2]).



(a) input sentence (b) prior

(c) after source-to-target features

(d) composed with n-gram

Figure 2: System design without $M - N$ alignment. The CRF system is realized using finite state automata. The input is modeled as a chain of input symbols `a`, `b`, `c` Fig. 2(a). This chain is augmented with the prior saved in a single state Fig. 2(b) and source-to-target features are applied to the automaton resulting in Fig. 2(c). Finally, the automaton is composed with the automaton sketched in Fig. 2(d) utilizing Φ/failure arcs (cf. [9, 10]).

Linear Chain Conditional Random Fields are convex functions, but with the use of a hidden variable the convexity is no longer guaranteed. Thus, the systems tends to get stuck in local optima. One solution to overcome this issue is to initialize the model with some convex model to prefer reasonable alignment paths. In this publication, the features corresponding to the current word (y_n, x_{a_n}) have been initialized with an IBM-1 model [15] trained on the same data utilizing giza++ [4].

4. M-to-N Alignments

Although it is possible to model alignments using HCRFs, they are unable to produce target sequences which are longer than the corresponding source sequences. The summation only assigns one target word to multiple source words.

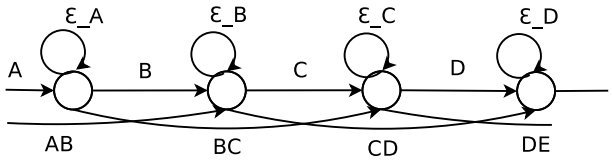


Figure 3: Hidden Markov Model assigning two target symbols on skip arcs (AB, BC, CD, DE). Forward arcs assign one target symbol per arc (A, B, C, D), and loop arcs assign no new target symbol (represented by named epsilons).

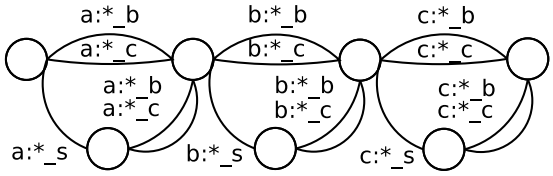


Figure 4: The M-N Alignment permits to assign one target symbol per source word (begin "_b"/continue "_c" tags) or two target symbols per source word (skip "_s" tags followed by begin/continue tags).

Classical HMM-Alignments instead can utilize skip arcs, which are used to skip one target symbol in the reference sequence (as e.g. within the so-called 0-1-2 standard model, which allows for loops, forward and skip transitions as frequently used in automatic speech recognition). Instead of skipping or removing a target symbol, a similar effect can be achieved by mapping a source symbol to two target symbols (cf. Fig. 3).

Fig. 4 shows a combination of the HCRFs with source symbol duplication. It is expected that if the source symbol doubling (skip) path is chosen, at least at the skip arc a new target symbol has been started.

Usually, penalties are introduced within HMMs for the different transition lengths (forward, loop and skip). These penalties are commonly chosen empirically. Within CRFs, it is possible to introduce three features which are active at the beginning δ_1, the continuation δ_0, and the duplication of a target symbol δ_2 trained within the CRF framework.

The final automaton represents the denominator in Eq. 3. The numerator of Eq. 3 is selected by a composition with an automaton representing all paths with respect to the given reference y_1^N. To avoid the distribution of the probability mass over too many paths, a simplified automaton could be applied, which allows for a skip arc to be followed by a continue arc and thus permitting overlapping alignments on source and target side.

5. Experimental Results

In this section, evidence of the performance of the proposed additions to the CRF approach is presented on the English Celex corpus [16]. The Celex corpus has an output vocabulary of 53 symbols (phonemes) and 39995 training samples. All presented experiments have been performed in the same framework, where some parameters have been kept fixed for all experiments: RProp is used as optimization algorithm [17], whereas the CRFs are always trained for 50 iterations. The regularization parameters for elastic net have been fixed to $c_1 = c_2 = \frac{1}{16}$. As error rates, the Levenshtein based phoneme error rate (PER) and the word error rate (WER) are used.

Table 1: Should a 0-1-2 Standard HMM or a simplified automaton permitting to continue target symbols after a skip symbol be used for numerator selection?

extract numerator	PER[%] Dev	PER[%] Eva	WER[%] Dev	WER[%] Eva
standard HMM	2.8	2.8	14.3	13.9
simplified	2.6	2.6	12.8	12.4

Table 2: Effect of transition penalties (cf. Sec. 4). "Empirically" is an experiment utilizing downhill simplex for tuning the δ_j, while "automatically" integrates the features in CRF training.

HMM weights tuning	PER[%] Dev	PER[%] Eva	WER[%] Dev	WER[%] Eva
none ($\delta_j = 0$)	25.5	25.7	86.1	86.2
empirically	2.9	2.9	14.9	13.9
automatically	2.6	2.6	12.8	12.4

As a baseline, experiments using an external alignment have been used (cf. Tab. 3). The alignment was generated on the training and test sets by the method proposed in [18] and used with Linear Chain CRFs. The training and recognition system was the same as used in the $M - N$ alignment experiments except the use of the source symbol doubling, summation in numerator (HCRF), and δ-features.

In Sec. 4 it has been discussed that both, a full 0-1-2 Standard HMM or a simplified automaton permitting to continue target symbols after a skip symbol can be used for extracting the numerator. It seems to be beneficial to use the simplified automaton resulting in a small improvement in PER and a significant improvement in WER (cf. Tab. 1).

In Tab. 2, it is shown that the system improves if the forward-/loop-/skip-penalties are included as features in the CRF training, i.e. when the penalties are trained automatically. If no weights are used, the error rates degrade badly. Interestingly, an empirical tuning on the dev set using the downhill simplex algorithm seems to be unable to find the optimal parameters, but is at least quite close to the best result.

The final system is composed of IBM-1 initialization,

Table 3: Feature Build-Up on Celex (cf. Sec. 5)

	PER[%]		WER[%]	
	Dev	Eva	Dev	Eva
[19]				10.8
joint n-grams [18]		2.5		11.4
external alignment from joint n-grams	2.8	2.8	13.5	13.5
(e_{a_j}, f_j) + (δ_j)	52.6	52.7	97.5	97.9
+ 2-grams	23.2	23.4	76.6	76.8
+ source n-grams	2.6	2.6	12.8	12.4
+ 3-grams	2.6	2.5	12.4	11.8

$M - N$ alignments, numerator extraction with a simplified HMM, δ-features, source n-grams, and target 3-grams. Using $M - N$ alignments in CRFs leads to better performance w.r.t. PER and WER.

6. Conclusion

The best error rate of 2.5% PER on the test set is the same as for the joint n-gram method proposed in [18]. Only the WER differs which can be accounted to a different distribution of the same amount of errors. Thus, it was possible to implement a CRF system which is able to produce results equal to the best published generative system for G2P conversion on this task. The paper [19] does not provide PER results only WER results. The discriminative training framework in [19] is similar to the CRFs shown in this publication, except the use of the MIRA algorithm and joint n-grams. We expect that the difference in WER can be accounted to missing of these joint n-gram features.

Finally, results comparable to the best state-of-the-art systems could be achieved with an CRF system trained on unaligned data.

7. Acknowledgements

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

8. References

[1] T. Lavergne, O. Cappé, and F. Yvon, "Practical Very Large Scale CRFs," in *Proceedings the 48th Annual Meeting of the Association for Computational Linguistics (ACL)*. Association for Computational Linguistics, July 2010, pp. 504–513.

[2] S. Hahn, M. Dinarelli, C. Raymond, F. Lefevre, P. Lehnen, R. De Mori, A. Moschitti, H. Ney, and G. Riccardi, "Comparing stochastic approaches to spoken language understanding in multiple languages," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. PP, no. 99, p. 1, 2010.

[3] T. Deselaers, S. Hasan, O. Bender, and H. Ney, "A Deep Learning Approach to Machine Transliteration," in *Proceedings of the EACL 2009 Workshop on Statistical Machine Translation*, Athens, Greece, Mar. 2009, pp. 233–241.

[4] F. J. Och and H. Ney, "A Systematic Comparison of Various Statistical Alignment Models," *Computational Linguistics*, vol. 29, no. 1, pp. 19–51, Mar. 2003.

[5] P. Blunsom and T. Cohn, "Discriminative Word Alignment with Conditional Random Fields," in *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics (ACL)*. Morristown, NJ, USA: Association for Computational Linguistics, Jul. 2006, pp. 65–72.

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

[7] 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, Jun. 2001, pp. 282–289.

[8] S. Hahn, P. Lehnen, and H. Ney, "Powerful extensions to CRFs for Grapheme to Phoneme Conversion," in *Proceedings of the Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2011.

[9] C. Allauzen, M. Mohri, and B. Roark, "Generalized algorithms for constructing statistical language models," in *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*, ser. ACL '03. Stroudsburg, PA, USA: Association for Computational Linguistics, 2003, pp. 40–47.

[10] P. Lehnen, S. Hahn, and H. Ney, "N-grams for conditional random fields or a failure-transition posterior for acyclic fsts," in *Interspeech*, Florence, Italy, Aug. 2011.

[11] A. Quattoni, S. Wang, L.-P. Morency, M. Collins, and T. Darrell, "Hidden Conditional Random Fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, pp. 1848–1852, 2007.

[12] T. Koo and M. 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: Association for Computational Linguistics, 2005, pp. 507–514.

[13] X. Yu and W. Lam, "Hidden Dynamic Probabilistic Models for Labeling Sequence Data," in *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence*, Chicago, IL, USA, Jul. 2008, pp. 739–745.

[14] P. Lehnen, S. Hahn, A. Guta, and H. Ney, "Incorporating alignments into conditional random fields for grapheme to phoneme conversion," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Prague, Czech Republic, May 2011, pp. 4916–4919.

[15] P. F. Brown, V. J. Della Pietra, S. A. Della Pietra, and R. L. Mercer, "The mathematics of statistical machine translation: Parameter estimation," *Computational Linguistics*, 1993.

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

[17] M. Riedmiller and H. Braun, "A direct adaptive method for faster backpropagation learning: The Rprop algorithm," in *IEEE International Conference on Neural Networks (ICNN)*, San Francisco, CA, USA, March – April 1993, pp. 586 – 591.

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

[19] S. Jiampojamarn, C. Cherry, and G. Kondrak, "Integrating joint n-gram features into a discriminative training framework," in *In Proceeding of the 11th Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT)*, Jun. 2010, pp. 697–700.