Conditional Random Field for tracking user behavior based on his eye's movements[1]

Trinh Minh Tri Do
LIP6, Université Paris 6
8 rue du capitaine Scott
75015, Paris, France
do@poleia.lip6.fr

Thierry Artières
LIP6, Université Paris 6
8 rue du capitaine Scott
75015, Paris, France
Thierry.artieres@lip6.fr

Abstract

Conditional Random Fields offer some advantages over traditional models for sequence labeling. These conditional models have mainly been introduced up to now in the information retrieval context for information extraction or POS-tagging tasks. This paper investigates the use of these models for signal processing and segmentation. In this context, the input we consider is a signal that is represented as a sequence of real-valued feature vectors and the training is performed using only partially labeled data. We propose a few models for dealing with such signals and provide experimental results on the data from the eye movement challenge.

1 Introduction

Hidden Markov models (HMM) have long been the most popular technique for sequence segmentation, e.g. identifying the sequence of phones that best matches a speech signal. Today HMM is still the core technique in most of speech engines or handwriting recognition systems. However, HMM suffer two major drawbacks. First, they rely on strong independence assumptions on the data being processed. Second, they are generative models that are most often learned in a non discriminant way. This comes from their generative nature, since HMM define a joint distribution $P(X,Y)$ over the pair of the input sequence (observations) X and the output sequence (labels) Y. Recently, conditional models including Maximum Entropy Markov models [1] and Condition Random Fields [2] have been proposed for sequence labeling. These models aim at estimating the conditional distribution $P(Y/X)$ and exhibit, at least in theory, strong advantages over HMMs. Being conditional models, they do not assume independence assumptions on the input data, and they are learned in a discriminant way. However, they rely on the manual and careful design of relevant features.

[1] This work was supported in part by the IST Program of the European Community, under the PASCAL Network of Excellence, IST-2002-506778.

Conditional Random Fields (CRF) has been shown to overcome traditional Markovian models in a series of information retrieval tasks such as information extraction, named entity recognition… Yet, CRF have to be extended to more general signal classification tasks. Indeed, the information retrieval context is very specific, considering for instance the nature of the input and output data. Designing relevant features for such data is maybe easier than for many other data. Also, algorithms proposed for training CRF require a fully labeled training database. This labeling may be available in information retrieval tasks since there is a kind of equivalence between nodes and labels but it is generally not available in other signal processing tasks. Hence, the previous usages of CRF do not fit well with many sequence classification and segmentation tasks concerning signals such as speech, handwriting etc. Input data is rougher; it is a sequence of real-valued feature vectors without precise semantic interpretation. Defining relevant features is then difficult. Also, training databases are not fully labeled. In speech and handwriting recognition for instance, data are labeled, at best, at the unit (phoneme or letter) level while it is often desirable to use a number of states for each unit, or even a number of modalities for a same unit (e.g. allograph in handwriting recognition).

This paper investigates the use of CRF models for such more general signal classification and segmentation tasks. We first introduce CRF and an extension called segmental CRF. Then we describe how to use CRF for dealing with multimodal classes and signal data and discuss corresponding inference and training algorithms. At last, we report experimental results concerning the eye movement challenge.

2 Conditional Random Fields for sequential data

Sequence labeling consists in identifying the sequence of labels $Y = y_1,...,y_T$ that best matches a sequence of observations $X = x_1...x_T$: $Y^* = \arg\max_Y P(Y/X)$. CRF are a particular instance of random fields. Figure 1 illustrates the difference between traditional HMM models and CRF. HMM (Fig. 1-a) are directed models where independence assumptions between two variables are expressed by the absence of edges. *CRF* are undirected graphical models, Figure 1-b shows a CRF with a chain structure. One must notice that CRF being conditional models, node *X* is observed so that *X* has not to be modeled. Hence CRF do not require any assumption about the distribution of *X*. From the random field theory, on can show [2] that the likelihood $P(Y/X)$ may be parameterized as:

$$P(Y/X,W) = \frac{e^{W.F(X,Y)}}{\sum_{Y'} e^{W.F(X,Y')}} = \frac{e^{W.F(X,Y)}}{Z_W(X)} \quad (1)$$

Where $Z_W(X) = \sum_{Y'} e^{W.F(X,Y')}$ is a normalization factor, $F(X,Y)$ is a feature vector and *W* is a weight vector. Features *F(X,Y)* are computed on maximal cliques of the graph. In the case of a chain structure (Fig. 1-b), these cliques are edges and vertices (i.e. a vertice y_t or an edge (y_{t-1}, y_t)).

In some cases there is a need to relax the Markovian hypothesis by allowing the process not to be Markovian within a state. [3] proposed for this semi-Markov CRF (SCRF). The main idea of these models is to use segmental features, computed on a segment of observations associated to a same label (i.e. node). Consider a segmentation of an input sequence $X = x_1, x_2,...,x_T$, this segmentation may be described as a sequence of segments $S = s_1, s_2,...,s_J$, with $J \le T$ and $s_j = (e_j, l_j, y_j)$

where e_j (l_j) stands for the entering (leaving) time in state (i.e. label) y_j. Segmental features are computed over segments of observations $x_{e_j},...,x_{l_j}$ corresponding to a particular label y_j. SCRF aims at computing $P(S/X)$ defined as in Eq. (1). To enable efficient dynamic programming, one assumes that the features can be expressed in terms of functions of X, s_j and y_{j-1}, these are segmental features:

$$F(X,S)=\sum_{j=1}^{|S|}F(X,y_{j-1},s_j)=\sum_{j=1}^{|S|}F(X,y_j,y_{j-1},e_j,l_j) \quad (2)$$

Inference in CRF and SCRF is performed with dynamic programming like algorithm. Depending on the underlying structure (chain, tree, or anything else) one can use Viterbi, Belief Propagation [4] or Loopy Belief Propagation [5]. Training in CRF consists in maximizing the log-likelihood *L(W)* based on a fully labeled database of *K* samples, $BA=\{(X_k,Y_k)\}_{k=1}^{K}$, where X_k is a sequence of observations and Y_k is the corresponding sequence of labels.

$$L(W)=\sum_{k=1}^{K}\log P(Y_k/X_k,W)=\sum_{k=1}^{N}((W.F(X_k,Y_k)-\log Z_W(X_k)) \quad (3)$$

This convex criterion may be optimized using gradient ascent methods. Note that computing $Z_W(X)$ includes a summation over an exponential number of label sequences that may be computed efficiently using dynamic programming. Training SCRF is very similar to CRF training and also relies on a fully labeled database.

(a) (b)

Figure 1: Dynamic representation of HMM (a) and CRF (b) as graphical models, where grey nodes represent observed variables.

3 Semi-Markov CRF for signal segmentation

We investigate the use of segmental CRF for signal segmentation. When dealing with real signals, one has to consider the continuous nature of input data, the multimodality of the classes and one has to develop algorithms for learning models without a fully labeled dataset. Hence, in the following we will consider that, during training, the label Y_k corresponding to input sequence X_k consists in the sequence of classes in X_k, whatever the length of the segments associated to these classes.

To take into account multimodality (e.g. a letter may be written with different styles) we investigate the use of a few states in a Segmental CRF model for each class, each one corresponds to a modality of the class. We will note *K* the number of states sharing the same label. Since there are several states corresponding to the same label, there are a number of segmentations *S* that correspond to a particular label sequence *Y*. Following [6] we introduce hidden variables for multimodality and segmentation information and build upon their work to develop inference and training algorithms with incomplete data. Hence, when conditioned on an input *X* the likelihood of a label sequence *Y* is defined as:

$$P(Y/X) = \sum_{S \in S(Y)} P(S/X) = \frac{\sum_{S \in S(Y)} \sum_{M} e^{W.F(X,S,M)}}{\sum_{S'} \sum_{M'} e^{W.F(X,S',M')}} \quad (4)$$

Where *S(Y)* stands for the set of segmentations *S* (defined as in §2) corresponding to the sequence of labels *Y*, *M* denotes a sequence of hidden variables, with $m_i \in \{1,...,K\}$. The use of hidden variables *(S,M)* makes inference expensive:

$$Y^* = \arg\max_Y P(Y/X) = \arg\max_Y \left[\sum_{S \in S(Y),M} e^{W.F(X,S,M)} \right] \quad (5)$$

Where *Y*, *S* and *M* have the same length, say *T*. This expression cannot be computed with a dynamic programming routine since the maximum and sum operators cannot be exchanged. However, if one uses a Viterbi approximation where summation is replaced with the maximum operator, and one assumes that $e^{W.F(X,S,M)}$ may be factorized in a product of *T* independent terms then the double maximization may be computed efficiently. Hence we use:

$$Y^* \approx \arg\max_Y \left[\operatorname{Max}_{S \in S(Y),M} P(S,M/X,W) \right] \quad (6)$$

Training aims at maximizing the log-likelihood *L(W)*. Using Eq. (4), the derivative of the likelihood of the k^{th} training example is computed as:

$$\frac{\delta L_k(W)}{\delta w_v} = \sum_{S \in S(Y_k)} \sum_{M} P(S,M/Y_k,X_k,W).F_v(X_k,S,M) - \sum_{S'} \sum_{M'} P(S',M'/X_k,W).F_v(X_k,S',M') \quad (7)$$

This criterion is expressed in terms of expected values of the features under the current weight vector that are $E_{P(S,M/Y_k,X_k,W)} F_v(X_k,S,M)$ and $E_{P(S,M/X_k,W)} F_v(X_k,S,M)$. These terms may be calculated using a forward-backward like algorithm since the CRF is assumed to have a chain structure. Based on the chain structure of the models we used two types of features: local features (computed on vertices) $F^1(X,y_t,q_t,m_t)$ and transition features computed on edges, $F^2(X,y_t,y_{t-1},q_t,q_{t-1},m_t,m_{t-1})$.

4 Eye movement challenge data

Here is a quick description of the challenge and of the data for the competition 1 of the challenge, see [7] for more details. The eye movement challenge concerns implicit feedback for information retrieval. The experimental setup is as follows. A subject was first shown a question, and then a list of ten sentences (called titles), one of which contained the correct answer (C). Five of the sentences were known to be irrelevant (I), and four relevant for the question (R). The subject was instructed to identify the correct answer and then press 'enter' (which ended the eye movement measurement) and then type in the associated number in the next screen. There are 50 such assignments, shown to 11 subjects. The assignments were in Finnish, the mother tongue of the subjects. The objective of the challenge is, for a given assignment, to predict the correct classification labels (I, R, C) of the ten sentences (actually only those that have been viewed by the user) based on the eye movements

alone. The database is divided in a training set of 336 assignments and a test set (the validation set according to challenge terminology) of 149 assignments. The data of an assignment is in the form of a time series of 22-dimensional eye movement features derived from the ones generally used in eye movement research (see [7]), such as fixation duration, pupils diameter etc. It must be noticed that there is a 23^{th} feature that consists in the number of the title being viewed (between 1 and 10).

5 Experiments

We applied segmental CRF as those described in §3 to eye movement data. The aim is to label the ten titles with their correct labels (I, R, C). This may be done though segmenting an input sequence with a CRF whose states correspond to labels I, R and C. We investigated a number of models for this. All models have been trained with a regularized likelihood criterion in order to avoid over fitting [6]. These models work on vectors of segmental features computed over segments of observations. A simple way to define segmental feature vector would be the average feature vector over the observations of the segment. However, the average operator is not necessarily relevant. We used ideas in [7] to choose the most adequate aggregation operator (sum, mean or max) for each of the 22 features.

The first model is a simple one. It is a SCRF model with three nodes, one for class *R*, one for class *C* and one for class *I*. It works on segmental features where segments correspond to sequences where the user visits one particular title. There is no transition features, corresponding to the change from one title to another one. This model is called 3NL for 3 Nodes CRF with Local features only (no transition features) and 3NLT if transitions features are added. It must be noticed that since a title may be visited more than on time in an assignment it is desirable that the labeling algorithm be consistent, i.e. finds a unique label for every title. This is ensured, whatever the model used, by adding constraints in the decoding algorithm.

One can design more complicated models by distinguishing between the different visits of a same title. For example, one can imagine that a user who visits a title a second or a third time will not behave as he did the first time. Maybe he may take more time or quickly scan all the words in the title… Hence, we investigated the use of SCRF models with two or three states per class (I, R, C). In the two states models, a first state is dedicated to the first visit to a title of class R, C or I. The second state is dedicated to all other posterior visits to this title. When using 3 states per class, we distinguish among the first visit, the last visit and intermediate visits to a title. These models are named 6NL and 9NL depending on their number of states per class (2 or 3) if they make use of local features, and 6NLT and 9NLT if they make use of local and transition features.

Finally, we investigate the use of multimodal models. Going back to the first model 3NL, we consider the use of a few states per class, this time corresponding to different ways of visiting a title (there is no chronological constraints). Models are named 3N2ML for 3 states, 2 Modes per class, and Local features.

Table 1 reports experimental results for various SCRF-based models and for 4 additional systems. The first one is a benchmark HMM system. It works on the same input representation (feature vectors) and has the same number of states as there are nodes in the 6NL model. The three other systems are combination systems combine three classifiers votes.

A first comment about the results is that all SCRF models outperform the HMM system. Also, using more complex models is not systematically better useful. We investigated two ways for this, firstly by taking into account the number of the visit (increasing the number of states), secondly by taking into account multimodality

(increasing the number of modes). Using a few states per class in order to take into account the number of a visit of a title allows reaching up to 73% (9NL) while allowing multimodality leads to poorer results. Also, we did not succeed in using efficiently transition features, this is still under investigation. At last, voting systems did not improve much over singles classifiers although HMM and SCRF systems tend to be complimentary. There is certainly some room for improvements here. Note however that we observed more stability in the results of voting systems when training and testing on various parts of the database.

Table 1 – Comparison of various systems on the eye movement challenge task.

Technique	System's name	#states- #modes	Features	Accuracy (%)
SCRF	3NL	3 - 1	L	71
-	3N2ML	3 - 2	L	71.5
-	3NLT	3 - 1	L + T	68.9
-	6NL	6 - 1	L	71.8
-	9NL	9 - 1	L	73.2
-	9N2ML	9 - 2	L	70.8
-	9NLT	9 - 1	L + T	69.4
HMM	HMM	6 states	L	66.2
Combination of 3NL, 6NL, 9NL			L	72.1
Combination of HMM, 6NL, 9NL			L	72.3
Combination of HMM, 3NL, 6NL			L	71.9

6 Conclusion

We presented systems based on conditional random fields for signal classification and segmentation. In order to process signals such as eye movement, speech or handwriting, we investigated the use of segmental conditional random fields and introduced the use of hidden variables in order to handle partially labeled data and multimodal classes. Experimental results on the eye movement challenge data show that our CRF models outperform HMM, but all results are rather close showing the difficulty of the task.

References

[1] McCallum, A., Freitag, D., and Pereira, F. (2000) Maximum entropy Markov models for information extraction and segmentation. *In Proc. ICML.*

[2] Lafferty, J., McCallum, A., and Pereira, F. (2001) Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *International Conf. on Machine Learning*, 282–289. Morgan Kaufmann, San Francisco, CA.

[3] Sarawagi, S., and Cohen, W. (2004) Semi-Markov Conditional Random Fields for Information Extraction. *Advances in Neural Information Processing Systems*.

[4] Weiss, Y. (2001) Correctness of belief propagation in Gaussian graphical models of arbitrary topology. *Neural Computation*, 13:2173-2200.

[5] Murphy, K., Weiss, Y., and Jordan, M. (1999) Loopy belief propagation for approximate inference: an empirical study. *In Proc. of the Conf. on Uncertainty in AI.*

[6] Quattoni A., Collins M. and Darrel T. (2004). Conditional Random Fields for Object Recognition. In Advances in Neural Information Processing Systems 17.

[7] Salojärvi, J., Puolamäki, K., Simola, J., Kovanen, L., Kojo, I., Kaski, S. (2005) Inferring Relevance from Eye Movements: Feature Extraction. Helsinki University of Technology, *Publications in Computer and Information Science*, Report A82.