PREDICTION OF F0 CONTOURS FROM SYMBOLIC AND NUMERICAL VARIABLES USING CONTINUOUS CONDITIONAL RANDOM FIELDS

Raul Fernandez [1], *Steve Minnis* [2], *Bhuvana Ramabhadran* [1]

[1] IBM TJ Watson Research Center, Yorktown Heights, NY
[2] Nuance Communications, Inc., Norwich, UK.
{fernanra,bhuvana}@us.ibm.com, steve.minnis@nuance.com

ABSTRACT

Regression of continuous-valued variables as a function of both categorical and continuous predictors arises in some areas of speech processing, such as when predicting prosodic targets in a text-to-speech system. In this work we investigate the use of Continuous Conditional Random Fields (CCRF) to conditionally predict F0 targets from a series of s symbolic and numerical predictive features derived from text. We derive the training equations for the model using a Least-Squares-Error criterion within a supervised framework, and evaluate the proposed system using this objective criterion against other baseline models that can handle mixed inputs, such as regression trees and ensemble of regression trees.

Index Terms— conditional regression, F0 prediction, speech synthesis

1. INTRODUCTION

In most text-to-speech (TTS) system architectures, a crucial component is responsible for making predictions about what a given prosodic property of a string of text should be, whether to generate a driving target that can guide the search through an inventory of units in a unit-selection system, or to use that prediction directly as a parameter of the output synthesis. Such predictions need to be made by examining features derived from the text, typically the only source of input available at run time, and usually consist of a series of symbolic attributes (such as syntactical and lexical properties) as well as numerical attributes (such as positional features) extracted by a text-analysis module. For real-valued prosodic variables, the modeling task is one of continuous regression from a heterogeneous set of symbolic and numeric regressors, a problem for which regression trees provide a well-known solution. In this work, we investigate an alternative model, continuous conditional random fields (CCRFs), that can be applied to this task. CCRFs directly provide a conditional distribution over the sequence of interest, rather than a joint distribution over both output sequence and input features, and therefore avoid having to model the input distribution directly, which in the case of heterogeneous symbolic and numeric inputs might prove difficult. Though CCRFs have been used for document ranking [1], tag recommendation [2] and remote-sensing [3] problems, their application in speech and prosody-modeling tasks remains unexplored. In Section 2 we review CCRFs and derive the training equations for the particular parametrization we use in this work using a least-squares approach. In Section 3 we cover their applicability to F0-modeling tasks and finally present some experimental results in Section 4.

2. CONTINUOUS CONDITIONAL RANDOM FIELDS

Let $\mathbf{y} = [y_1, \cdots, y_T]^T$ be a set of real-valued random variables associated with an arbitrary input sequence $\mathbf{x} = \{\mathbf{x}_1, \cdots, \mathbf{x}_T\}$ on which we have defined a set of K real-valued feature functions $\{f^{(1)}(y_t, \mathbf{x})\}_{k=1}^{K}$, and a set of L real-valued feature functions on pairwise observations $\{f^{(2)}(y_{t_1}, y_{t_2}, \mathbf{x})\}_{l=1}^{L}$. A Continuous Conditional Random Field is a *conditional* probability density function with the following general log-linear form [1]

$$p(\mathbf{y}|\mathbf{x};\alpha,\beta) = \frac{1}{Z(\mathbf{x})}\exp\{E(\mathbf{x},\mathbf{y};\alpha,\beta)\} \quad (1)$$

$$E(\mathbf{x},\mathbf{y};\alpha,\beta) = \sum_{t=1}^{T}\sum_{k=1}^{K}\alpha_k f_k^{(1)}(y_t,\mathbf{x}) + \sum_{t_1,t_2=1}^{T}\sum_{l=1}^{L}\beta_l f_l^{(2)}(y_{t_1},y_{t_2},\mathbf{x}) \quad (2)$$

$$Z(\mathbf{x}) = \int_{\mathbb{R}^T}\exp\{E(\mathbf{x},\mathbf{y};\alpha,\beta)\}d\mathbf{y} \quad (3)$$

where α and β are parameters in the model, and $Z(\mathbf{x})$ is the partition function ensuring a proper probability distribution. In this work we interpret the indices $t = \{1, \cdots, T\}$ associated with variables $\mathbf{x}$ and $\mathbf{y}$ to correspond to a temporally-ordered time series (though in general this need not be the case). The model in Eqns. 1-3 can be used to predict an observation sequence, given a set of input predictors $\mathbf{x}$, as the maximum a-posteriori estimate $\hat{\mathbf{y}} = \arg\max_{\mathbf{y}} p(\mathbf{y}|\mathbf{x})$. For arbitrary choices of $f^{(1)}$ and $f^{(2)}$, however, the computation of the partition function needed for inference and prediction is not tractable. Though sampling methods, for instance, may be used, we consider instead the following parametrization of the model in terms of quadratic feature functions that leads to a posterior in the Gaussian family, and therefore to an analytically computable MAP estimate:

$$E(\mathbf{x},\mathbf{y};\theta) = -\sum_{t}^{T}\sum_{r}^{R}\sum_{k}^{K}\alpha_{rk}M_r(\mathbf{x}_t)[y_t - \gamma_k h_k(\mathbf{x}_t)]^2 - \sum_{t_1,t_2}^{T}\sum_{l}^{L}\beta_l s_l(\mathbf{x}_{t_1},\mathbf{x}_{t_2})[y_{t_1} - y_{t_2}]^2, \quad (4)$$

where $\{h_k(\mathbf{x}_t)\}_{k=1}^{K}$ is a set of K real-valued functions of the input at time t; $\{M_r(\mathbf{x}_t)\}_{r=1}^{R}$ is a set of Boolean feature functions indicating when the input at time t belongs to one of R nominal categories; $\{s_l(\mathbf{x}_{t_1},\mathbf{x}_{t_2})\}_{l=1}^{L}$ is a set of L real-valued feature functions that measure proximity or closeness between two points in input space; and $Z(\mathbf{x})$ is the corresponding partition function. Be-

sides the claimed tractability, notice that this parametrization automatically incorporates a way to handle nominal- and real-valued properties of the input via the $M_r(\cdot)$ and $h_k(\cdot)$ functions respectively, a property that is useful when dealing with regression problems that make use of a mix of nominal and real-valued regressors. The parameters θ of the model are the weights α_{rk} and β_l, as well as the scale factors γ_k, leading to a parameter dimensionality of $RK + K + L$. To highlight the two-level component of this model, we will also refer to $M_r(\cdot)$, $h_k(\cdot)$ and $s_l(\cdot)$ as the α-Indicators, α-Features and β-Features respectively. The first term in 4 has the function of coupling each output to the input sequence whereas the second term couples pairs of outputs and the input. The CCRF can be viewed as a graphical model with edges between these pairs of nodes, and though in general the second sum in Eq. 4 ranges over $1 \leq t_2 \leq T$ (defining a fully-connected graphical model with edges between any pair of output variables), we are interested in a P-order CCRF where we assume the β-Features are zero outside the range $\max(t_1 - P, 1) \leq t_2 \leq \min(t_1 + P, T)$, leading to some sparsity.

After expanding and collecting terms by powers of y_t, Eq. 4 can be expressed as

$$E(\mathbf{x}, \mathbf{y}; \theta) = -\mathbf{y}'(Q + D - S)\mathbf{y} + 2\mathbf{b}'\mathbf{y} + c, \tag{5}$$

where Q and D are $T \times T$ diagonal matrices, S a $T \times T$ matrix, $\mathbf{b}$ a $T \times 1$ vector, and c a constant, with entries defined as follows for $i, j \in [1, T]$:

$$Q^{[ii]} = \sum_{r=1}^{R} M_r(\mathbf{x}_i) \sum_{k=1}^{K} \alpha_{rk} \tag{6}$$

$$D^{[ii]} = \sum_{l=1}^{L} \beta_l \sum_{j=1}^{T} s_l(\mathbf{x}_i, \mathbf{x}_j) \tag{7}$$

$$S^{[ij]} = \sum_{l=1}^{L} \beta_l s_l(\mathbf{x}_i, \mathbf{x}_j) \tag{8}$$

$$\mathbf{b}^{[i]} = \sum_{k=1}^{K} \gamma_k h_k(\mathbf{x}_t) \sum_{r=1}^{R} \alpha_{rk} M_r(\mathbf{x}_i) \tag{9}$$

$$c = -\sum_{t=1}^{T} \sum_{k=1}^{K} \sum_{r=1}^{R} \alpha_{rk} \gamma_k^2 M_r(\mathbf{x}_t) h_k^2(\mathbf{x}_t). \tag{10}$$

From Eq. 5 we can recognize that the numerator of Eq. 1 is in the form of an unnormalized Gaussian, and that $Z(\mathbf{x})$ is the normalizing Gaussian integral which, provided we impose the constraints $\alpha_{rk}, \beta_l > 0$ for the integral to exist, can be evaluated to be

$$Z(\mathbf{x}) = (2\pi)^{\frac{T}{2}} |\Sigma|^{\frac{1}{2}} \exp\left\{ \frac{\mu' \Sigma^{-1} \mu}{2} + c \right\}, \tag{11}$$

with $\mu = (Q + D - S)^{-1}\mathbf{b}$ and $\Sigma = \frac{1}{2}(Q + D - S)^{-1}$, leading to a Gaussian conditional posterior $p(\mathbf{y}|\mathbf{x}; \theta) = \mathcal{N}(\mathbf{y}; \mu(\mathbf{x}), \Sigma(\mathbf{x}), \theta)$ and therefore to a MAP estimate $\hat{\mathbf{y}}(\mathbf{x}; \theta) = (Q + D - S)^{-1}\mathbf{b}$ (where we have made the data-dependency explicit this once to highlight the conditional nature of the model).

2.1. Training

To train a CCRF, we adopt an L_2 prediction (or generation) error minimization framework to make the training phase consistent with the mean-square error (MSE) criterion we will use to objectively evaluate the predictions. That is, for a set of training data $\{\mathbf{x}, \mathbf{y}\}_{n=1}^{N}$ with N sequences, each of length T_n, we seek to minimize

$$g(\theta) = \frac{1}{2} \sum_{n}^{N} \mathbf{e}_n' \mathbf{e}_n = \frac{1}{2} \sum_{n}^{N} (A_n^{-1}\mathbf{b}_n - \mathbf{y}_n)'(A_n^{-1}\mathbf{b}_n - \mathbf{y}_n) \tag{12}$$

subject to $\alpha_{rk}, \beta_l > 0$ (using the shorthand $A = Q + D - S$). Since we know no closed-form solution to this problem, we implement gradient-descent techniques using a log barrier on α_{rk} and β_l to ensure strict positivity. The quantities of interest, then, are the components of the gradient vector ∇_θ given by

$$\begin{aligned}
\frac{\partial g(\theta)}{\partial \log \alpha_{rk}} &= \alpha_{rk} \frac{\partial g(\theta)}{\partial \alpha_{rk}} \\
&= \alpha_{rk} \sum_{n}^{N} \mathbf{e}_n' \times \left\{ A_n^{-1} \frac{\partial \mathbf{b}_n}{\partial \alpha_{rk}} - A_n^{-1} \frac{\partial Q_n}{\partial \alpha_{rk}} A_n^{-1} \mathbf{b}_n \right\} \\
&= \alpha_{rk} \Bigg\{ \sum_{n}^{N} (A_n^{-1}\mathbf{b}_n - \mathbf{y}_n)' A_n^{-1} \times \\
&\qquad (\mathbf{w}_{rk_n} - L_{r_n} A_n^{-1} \mathbf{b}_n) \Bigg\}
\end{aligned} \tag{13}$$

where we have made use of the following: $\frac{\partial Q_n}{\partial \alpha_{rk}} = L_{r_n}$ and $\frac{\partial \mathbf{b}_n}{\partial \alpha_{rk}} = \mathbf{w}_{rk_n}$, with L_{r_n} a $T_n \times T_n$ diagonal matrix and $\mathbf{w}_{rk_n}$ a $T_n \times 1$ vector with respective entries $L_r^{[ii]} = M_r(\mathbf{x}_i)$ and $\mathbf{w}_{rk}^{[i]} = \gamma_k M_r(\mathbf{x}_i) h_k(\mathbf{x}_i)$. Likewise,

$$\begin{aligned}
\frac{\partial g(\theta)}{\partial \log \beta_l} &= \beta_l \frac{\partial g(\theta)}{\partial \beta_l} \\
&= \beta_l \sum_{n}^{N} \mathbf{e}_n' \left(-A_n^{-1} \frac{\partial (D_n - S_n)}{\partial \beta_l} A_n^{-1} \mathbf{b}_n \right) \\
&= -\beta_l \sum_{n}^{N} (A_n^{-1}\mathbf{b}_n - \mathbf{y}_n)' A_n^{-1} R_{l_n} A_n^{-1} \mathbf{b}_n,
\end{aligned} \tag{14}$$

with $R_{l_n} = \frac{\partial D_n}{\partial \beta_l} - \frac{\partial S_n}{\partial \beta_l}$ a $T_n \times T_n$ matrix with entries

$$R_l^{[ij]} = \begin{cases} \sum_{m \neq i} s_l(\mathbf{x}_i, \mathbf{x}_m) & : i = j \\ -s_l(\mathbf{x}_i, \mathbf{x}_j) & : i \neq j. \end{cases} \tag{15}$$

Finally, the unconstrained gradient with respect to γ_k is given by

$$\frac{\partial g(\theta)}{\partial \gamma_k} = \sum_{n}^{N} \mathbf{e}_n' A_n^{-1} \frac{\partial \mathbf{b}_n}{\partial \gamma_k} \tag{16}$$

$$= \sum_{n}^{N} (A_n^{-1}\mathbf{b}_n - \mathbf{y}_n)' A_n^{-1} \mathbf{u}_{k_n}, \tag{17}$$

with $\mathbf{u}_{k_n}$ a $T_n \times 1$ vector with entries $\mathbf{u}_k^{[i]} = h_k(\mathbf{x}_i) \sum_r \alpha_{rk} M_r(\mathbf{x}_i)$.

Eqns. 13-17 were used to minimize Eq. 12 using the limited-memory BFGS quasi-Newton gradient-descent method and the libLBFGS software library implementation [4]. Step size was determined at every iteration by performing a line-search that met the strong Wolfe conditions, and the optimization stopped after a maximum of 20 line-search iterations.

3. FEATURES

Since we are interested in the application of CCRFs to the automatic generation of F0 contours for text-to-speech (TTS) applications, we limit the predictor features to those that can be extracted from text analysis of an input text sequence. Given a dual corpus of text and corresponding audio waveforms, a TTS front-end (FE) is used to analyze the input text to carry out the usual tokenization, normalization and baseform generation, and this information is used to force-align the audio against the input phone sequences using 3-state, left-to-right hidden Markov models.

Based on the FE analysis, the following strictly hierarchical set of "structural units" is defined to help with feature definition: *phone*, *syllable*, *word*, *syntax group* (or *STX-Group*), *punctuation group* (or *P-Group*), and *sentence*. Outside this hierarchy, we also consider the following two levels: *stressed-phone group* (or *SP-Group*), and the *stressed-syllable group* or (*SS-Group*); as the names suggest, they refer to the span between adjacent stressed phones and syllables, respectively, and subsume only phones. All features are extracted and assigned at the state level; feature values defined at a broader unit are propagated down to constituent states (e.g., all states of all phones in a word would inherit the same word-level attribute).

The set of 35 α-Features include: (a) distance features: number of states to/from the nearest boundary of each structural domain (12 features) as well as to/from the nearest stressed phone and stressed syllable, leading to 16 distance features; (b) counts: number of sub-units subsumed by any structural unit (e.g., *#phones* in { *SP-Group, syllable, SS-Group, word, STX-Group, P-Group, sentence* }; *#S-Group* in {*P-Group, sentence*}, etc.), leading to 17 features; and (c) 2 estimates of word-level pitch-accent (PA) probability: one based on the PA-ratio introduced in [5], and one based on a text-to-PA predictor using the CRF-based system described in more detail in [6].

The set of 13 measurements that give rise to the α-Indicators[1] include Boolean functions encoding: canonical voicing status of a phone, consonant-vowel distinction, and word-level membership in 5 pre-defined broad lexical categories (*Function Words*, *WH Words*, *Auxiliary Verbs*, *Conjunctions*, and *Adpositions*). We also consider features encoding place of articulation (9 categories); type of post-lexical punctuation (6); syllable-level lexical stress (3); part-of-speech (35); and a 3-way named-entity feature. Additionally, we employ a novel feature, which gives rise to the *STX-Group* mentioned above. The STX-Group is designed to characterize the observed patterns found in grammatical surface form, where grammatical form is determined in chunked units, not dissimilar to those auto-extracted in statistical machine translation [7]. The Marker Hypothesis [8] is then used as a guide to categorize these chunks, with the option of further clustering of similar chunks based on patterns of simple non-recursive syntactic structures. This can reduce the final number of STX-Groups used in our approach (in this work, we use a set of size 25. Lexical-semantic features, such as those defined in our alpha-indicators, are designed to filter these STX-Groups so that members of a group are semantically homogeneous. In the future we envision using additional relevant features which describe the context of the STX-Groups, such as paragraph, sentence or list markup, as well as word class semantic similarity measures.

Finally, the β-Features $s_l(\mathbf{x}_{t_1}, \mathbf{x}_{t_2})$ in 4 are defined so as to encode a notion of proximity in input space, and use this quantity to weigh the output-difference term $(y_{t_1} - y_{t_2})^2$. Intuitively, this component of the model gives us a handle to control smoothness by downplaying output differences when these correspond to inputs that are farther from each other. A function that has this simple property is a symmetric Gaussian kernel of the form:

$$s_l(\mathbf{x}_{t_1}, \mathbf{x}_{t_2}) = \exp\left(\frac{I^l(t_2) - I^l(t_1)}{\sigma_l^2}\right), \qquad (18)$$

where $I^l(t)$ is a function associating an index across some dimension l to the sample at time t (e.g., the third phone of the second syllable in the first word, etc.). In this work, we measure this proximity along $L = 7$ different time scales (the state level, plus the 6-level hierarchy defined above) and use Eqn. 18 to map an absolute distance along each scale to the interval $[0, 1]$ with free scale parameter σ_l.

[1]Notice that R in Eqn. 4 is not the number of such measurements, but the overall cardinality of their value sets $R = \sum_j |V_j|$. In other words, there is an indicator $M_r(\cdot)$ for each value in the domain of each of these measurements.

4. EXPERIMENTS AND RESULTS

Experiments were conducted based on a dual corpus containing approximately 10 hours of read speech, aligned as previously described. $\log(F0)$ contours were extracted using Praat and interpolated to obtain a continuous curve throughout all non-silence regions. Since the data contains several speakers, the $\log(F0)$ observations were speaker-normalized to obtain speaker-independent $\log(F0)$ z-scores. The features described in Section 3 were extracted for each state-level unit and paired with the z-score value from the state's mid-point to form a dataset, of which 70% was used for training and 15% reserved for each of a development and testing sets.

We investigated the performance of CCRF models with respect to the baseline performance of regression trees (RTs) since they are widely-adopted models when dealing with regression from nominal and numerical predictors (and are commonly used in language and speech applications). Additionally, we investigated the performance of single models (in each model class) with respect to the performance of an ensemble since averaging ensemble methods often lead to a boosted performance over single models. For RTs, in particular, we adopted building randomized forests of regression trees [9], an ensemble methodology that has been shown to behave robustly to overfitting and exhibit state-of-the-art performance in many regression problems [10]. Both the single and random trees were grown subject to the following stopping criteria: a minimum number of 70 observations per leaf, and a minimum RMS tolerance per node of 10% (e.g., the percent of the training set's RMS). All trees were fully grown and then pruned, and a forest of 25 trees was grown considering only a random draw of 50% of all available variables at each node split. The feature set consisted of those predictors already described in Section 3 for each time sample (state), augmented by a context window of two time samples on either side.

To create a CCRF ensemble, 25 models were built by drawing with replacement a subset of the α-Indicators (4 randomly chosen out of 13), α-Features (7/35) and β-Features (4/7) available for each training sample (plus a context of two training samples on either side) when training each model. The scale parameter was randomly drawn from the set $\sigma_l = \{0.5, 1.0\}$. This process significantly reduces the dimensionality of the input space in which the optimization takes place (since this dimensionality increases as the product $R \times K$), and speeds up the learning process with respect to the training time of a single model built on all the features at once. We carried out only some preliminary experiments with CCRF order, selecting to build models of order $P = 6$ after seeing comparable results on the development set for orders $P = 3$ and 6.

Results of these experiments are reported on Table 1 for the test set. Looking at the rows of this table, we see a noticeable relative

	RegTree	CCRF	Relative Error Reduction
Single Model	.3236	.2790	+13.78%
Ensemble	.3194±.0013	.2864±.0017	+10.33%
Fusion	.2797	.2835	-1.3%
Fusion Relative Error Reduction	+13.57%	-1.61%	

Table 1. Absolute mean-squared error and relative error reduction between different models on the testing set.

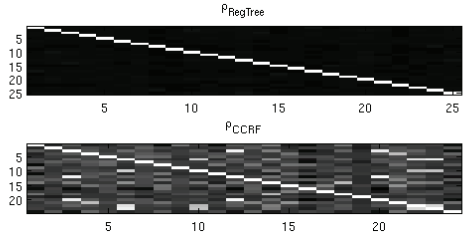

Fig. 1. Matrix of pair-wise cross-correlation coefficients ρ for tree and CCRF ensembles.

reduction in the MSE of a single, global CCRF model with respect to the single regression tree built on the full feature set (13.78%). When we look at the distribution of MSE across members of each ensemble, we also observe that, on the average, randomly-built CCRFs outperform randomly-built regression trees in relative error by 10.33%. However, when the outputs of each of these ensembles are fused by ensemble average, regression trees outperform the CCRF fused output by 1.3%. It would seem that in spite of lagging behind when it comes to single-model performance, the trees may be benefiting from higher complementarity among the outputs to produce a noticeable ensemble gain over single-model trees and ensemble CCRFs. To verify this, we examined the cross-correlation coefficient ρ_{ij} between the predictions $\hat{\mathbf{y}}_i$ and $\hat{\mathbf{y}}_j$ of any two members within each ensemble. This is plotted in Fig. 1 for $1 \leq i, j \leq 25$. The strongly diagonal structure of the top of this figure confirms that each of the outputs of the tree ensemble is strongly uncorrelated with the remaining ensemble outputs, so that system combination leads to a considerable relative improvement (13.57%). In contrast, the lower panel shows how the CCRF ensemble outputs strongly correlate with each other, so that no gains are observed when we try to exploit CCRFs in the ensemble scheme. Instead we see a drop with respect to the best single-model CCRF output of 1.61%. (This behavior was persistent even when we trained a model with mutually exclusive feature subsets to attempt to reduce redundancy in the input and hope for more uncorrelated outputs.) One weakness in the ensemble training of CCRFs is that there is no explicit criterion to ensure diversity among the different members: each predictor is trained independently to minimize MSE on the feature subset available to it at the beginning of training. Random trees, on the other hand, though also lacking such an explicit criterion, manage to (i) have access in principle, and therefore to potentially exploit, the full feature set during the entire training procedure while (ii) injecting randomization at every split by considering only a finite subset of this set. The CCRF training we implemented, however, limits the view of the data of each member from the start by training on a fixed subset. Our ongoing work focuses on modifying the training of ensemble models to directly incorporate diversity among them in the optimization criterion and investigate whether such techniques can lead to gains over single models and tree ensembles. Overall, the best numerical result was obtained with a single CCRF, though its performance is not significantly different from that of the fused output of a tree ensemble.

5. CONCLUSIONS

In this work we applied CCRFs to time-series prediction from categorical and numerical variables using a minimum MSE training criterion, and applied them to the task of modeling F0 contours from text with the goal of exploiting them in text-to-speech applications that can make use of a prosody-generation module. To our knowledge, the investigation of CCRFs for prosody modeling and speech synthesis, and the training procedure presented here, are novel contributions. We showed that individual CCRFs exhibit significant gains (of more than 10% relative MSE reduction) when compared with a single regression tree and with randomly grown regression trees, and that a single model's performance is comparable to that obtained with an ensemble of regression trees. When we investigated further enhancing this initial gain by using CCRFs within ensemble methods, however, we observed a high degree of correlation among member outputs, leading to small loss in performance over a single CCRF and tree ensembles. Our ongoing and future work is two-fold: to overcome this limitation by directly incorporating into the training procedure notions of diversity among ensemble members, and expanding the training criterion to incorporate other performance metrics, beyond MSE, that might be relevant to the particular domain of speech synthesis and that address known perceptually desirable properties of the output (such as smoothness, global variance, etc.).

6. REFERENCES

[1] T. Qin, et al., "Global ranking using continuous conditional random fields," in *Proc. NIPS*, 2008, pp. 1281–1288.

[2] X. Liu, et al., "Tag recommendation based on continuous conditional random fields," in *Proc. Intnl. Conf. Information Mgt., Innovation Mgt. and Industrial Eng.*, 2009, pp. 475–480.

[3] V. Radosavljevic, S. Vucetic, and Z. Obradovic, "Continuous conditional random fields for regression in remote sensing," in *Proc. (ECAI)*, 2010, pp. 809–814.

[4] N. Okazaki, "libLBFGS: a library of limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS)," http://www.chokkan.org/software/liblbfgs/.

[5] A. Nenkova, et al., "To memorize or to predict: Prominence labeling in conversational speech," in *Proc. HLT-ACL*, Rochester, NY, April 2007, pp. 9–16.

[6] R. Fernandez and B. Ramabhadran, "Driscriminative training and unsupervised adaptation for labeling prosodic events with limited training data," in *Interspeech*, 2010, pp. 1429–1432.

[7] P. Brown, et al., "A statistical approach to machine translation," *Computational Linguistics*, vol. 16, pp. 79–85, 1990.

[8] T. Green, "The necessity of syntax markers: Two experiments with artificial languages," *J. Verbal Learning and Behavior*, vol. 18, pp. 481–496, 1979.

[9] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[10] M.R. Segal, "Machine learning benchmarks and random forest regression," Tech. Rep., Center for Bioinformatics and Molecular Biostatistics, Univ. California, SF., 2004.