Incremental Feature Selection and ℓ_1 Regularization for Relaxed Maximum-Entropy Modeling

Stefan Riezler and **Alexander Vasserman**
Palo Alto Research Center
3333 Coyote Hill Road, Palo Alto, CA 94304

Abstract

We present an approach to bounded constraint-relaxation for entropy maximization that corresponds to using a double-exponential prior or ℓ_1 regularizer in likelihood maximization for log-linear models. We show that a combined incremental feature selection and regularization method can be established for maximum entropy modeling by a natural incorporation of the regularizer into gradient-based feature selection, following Perkins et al. (2003). This provides an efficient alternative to standard ℓ_1 regularization on the full feature set, and a mathematical justification for thresholding techniques used in likelihood-based feature selection. Also, we motivate an extension to n-best feature selection for linguistic features sets with moderate redundancy, and present experimental results showing its advantage over ℓ_0, 1-best ℓ_1, ℓ_2 regularization and over standard incremental feature selection for the task of maximum-entropy parsing.[1]

1 Introduction

The maximum-entropy (ME) principle, which prescribes choosing the model that maximizes the entropy out of all models that satisfy given feature constraints, can be seen as a built-in regularization mechanism that avoids overfitting the training data. However, it is only a weak regularizer that cannot avoid overfitting in situations where the number of training examples is significantly smaller than the number of features. In such situations some features will occur zero times on the training set and receive negative infinity weights, causing the assignment of zero probabilities for inputs including those features. Similar assignment of (negative) infinity weights happens to features that are pseudo-minimal (or pseudo-maximal) on the training set (see Johnson et al. (1999)), that is, features whose value on correct parses always is less (or greater) than or equal to their value on all other parses. Also, if large features sets are generated automatically from conjunctions of simple feature tests, many features will be redundant. Besides overfitting, large feature sets also create the problem of increased time and space complexity.

Common techniques to deal with these problems are regularization and feature selection. For ME models, the use of an ℓ_2 regularizer, corresponding to imposing a Gaussian prior on the parameter values, has been proposed by Johnson et al. (1999) and Chen and Rosenfeld (1999). Feature selection for ME models has commonly used simple frequency-based cut-off, or likelihood-based feature induction as introduced by Della Pietra et al. (1997). Whereas ℓ_2 regularization produces excellent generalization performance and effectively avoids numerical problems, parameter values almost never decrease to zero, leaving the problem of inefficient computation with the full feature set. In contrast, feature selection methods effectively decrease computational complexity by selecting a fraction of the feature set for computation; however, generalization performance suffers from the ad-hoc character of hard thresholds on feature counts or likelihood gains.

Tibshirani (1996) proposed a technique based on ℓ_1 regularization that embeds feature selection into regularization such that both a precise assessment of the reliability of features and the decision about inclusion or deletion of features can be done in the same framework. Feature sparsity is produced by the polyhedral structure of the ℓ_1 norm which exhibits a gradient discontinuity at zero that tends to force a subset of parameter values to be exactly zero at the optimum. Since this discontinuity makes optimization a hard numerical problem, standard gradient-based techniques for estimation cannot be applied directly. Tibshirani (1996) presents a specialized optimization algorithm for ℓ_1 regularization for linear least-squares regression called the Lasso algorithm. Goodman (2003) and Kazama and Tsujii (2003) employ standard iterative scaling and conjugate gradient techniques, however, for regulariza-

[1]This research has been funded in part by contract MDA904-03-C-0404 of the Advanced Research and Development Activity, Novel Intelligence from Massive Data program.

tion a simplified one-sided exponential prior is employed which is non-zero only for non-negative parameter values. In these approaches the full feature space is considered in estimation, so savings in computational complexity are gained only in applications of the resulting sparse models. Perkins et al. (2003) presented an approach that combines ℓ_1 based regularization with incremental feature selection. Their basic idea is to start with a model in which almost all weights are zero, and iteratively decide, by comparing regularized feature gradients, which weight should be adjusted away from zero in order to decrease the regularized objective function by the maximum amount. The ℓ_1 regularizer is thus used directly for incremental feature selection, which on the one hand makes feature selection fast, and on the other hand avoids numerical problems for zero-valued weights since only non-zero weights are included in the model. Besides the experimental evidence presented in these papers, recently a theoretical account on the superior sample complexity of ℓ_1 over ℓ_2 regularization has been presented by Ng (2004), showing logarithmic versus linear growth in the number of irrelevant features for ℓ_1 versus ℓ_2 regularized logistic regression.

In this paper, we apply ℓ_1 regularization to log-linear models, and motivate our approach in terms of maximum entropy estimation subject to relaxed constraints. We apply the gradient-based feature selection technique of Perkins et al. (2003) to our framework, and improve its computational complexity by an n-best feature inclusion technique. This extension is tailored to linguistically motivated feature sets where the number of irrelevant features is moderate. In experiments on real-world data from maximum-entropy parsing, we show the advantage of n-best ℓ_1 regularization over ℓ_2, ℓ_1, ℓ_0 regularization and standard incremental feature selection in terms of better computational complexity and improved generalization performance.

2 ℓ_{p} Regularizers for Log-Linear Models

Let $p_{\boldsymbol{\lambda}}(x|y) = \frac{e^{\sum_{i=1}^{n} \lambda_i f_i(x,y)}}{\sum_x e^{\sum_{i=1}^{n} \lambda_i f_i(x,y)}}$ be a conditional log-linear model defined by feature functions $\boldsymbol{f}$ and log-parameters $\boldsymbol{\lambda}$. For data $\{(x_j, y_j)\}_{j=1}^{m}$, the objective function to be minimized in ℓ_{p} regularization of the negative log-likelihood $L(\boldsymbol{\lambda})$ is

$$\begin{aligned} C(\boldsymbol{\lambda}) &= L(\boldsymbol{\lambda}) + \Omega_{\mathrm{p}}(\boldsymbol{\lambda}) \\ &= -\frac{1}{m} \sum_{j=1}^{m} \ln p_{\boldsymbol{\lambda}}(x_j|y_j) + \gamma \|\boldsymbol{\lambda}\|_{\mathrm{p}}^{\mathrm{p}} \end{aligned}$$

The regularizer family $\Omega_{\mathrm{p}}(\boldsymbol{\lambda})$ is defined by the Minkowski ℓ_{p} norm of the parameter vector $\boldsymbol{\lambda}$ raised to the p^{th} power, i.e. $\|\boldsymbol{\lambda}\|_{\mathrm{p}}^{\mathrm{p}} = \sum_{i=1}^{n} |\lambda_i|^{\mathrm{p}}$. The essence of this regularizer family is to penalize overly large parameter values. If $\mathrm{p} = 2$, the regularizer corresponds to a zero-mean Gaussian prior distribution on the parameters with γ corresponding to the inverse variance of the Gaussian. If $\mathrm{p} = 0$, the regularizer is equivalent to setting a limit on the maximum number of non-zero weights. In our experiments we replace ℓ_0 regularization by the related technique of frequency-based feature cutoff.

ℓ_1 regularization is defined by the case where $\mathrm{p} = 1$. Here parameters are penalized in the sum of their absolute values, which corresponds to applying a zero-mean Laplacian or double exponential prior distribution of the form

$$p(\lambda_i) = \frac{1}{2\tau} e^{-\frac{|\lambda_i|}{\tau}}$$

with $\gamma = \frac{1}{\tau}$ being proportional to the inverse standard deviation $\sqrt{2}\tau$. In contrast to the Gaussian, the Laplacian prior puts more mass near zero (and in the tails), thus tightening the prior by decreasing the standard deviation τ provides stronger regularization against overfitting and produces more zero-valued parameter estimates. In terms of ℓ_1-norm regularization, feature sparsity can be explained by the following observation: Since every non-zero parameter weight incurs a regularizer penalty of $\gamma|\lambda_i|$, its contribution to minimizing the negative log-likelihood has to outweigh this penalty. Thus parameters values where the gradient at $\lambda = 0$ is

$$\left| \frac{\partial L(\boldsymbol{\lambda})}{\partial \lambda_i} \right| \leq \gamma \qquad (1)$$

can be kept zero without changing the optimality of the solution.

3 Bounded Constraint Relaxation for Maximum Entropy Estimation

As shown by Lebanon and Lafferty (2001), in terms of convex duality, a regularization term for the dual problem corresponds to a "potential" on the constraint values in the primal problem. For a dual problem of regularized likelihood estimation for log-linear models, the corresponding primal problem is a maximum entropy problem subject to relaxed constraints. Let $H(p)$ denote the entropy with respect to probability function p, and $g : \mathbb{R}^n \to \mathbb{R}$ be a convex potential function, and $\tilde{p}[\cdot]$ and $p[\cdot]$ be expectations with respect to the empirical distribution $\tilde{p}(x, y) = \frac{1}{m} \sum_{j=1}^{m} \delta(x_j, x)\delta(y_j, y)$ and the

model distribution $p(x|y)\tilde{p}(y)$. The primal problem can then be stated as

$$\text{Maximize } H(p) - g(c) \text{ subject to}$$
$$p[f_i] - \tilde{p}[f_i] = c_i, i = 1, \ldots, n$$

Constraint relaxation is achieved in that equality of the feature expectations is not enforced, but a certain amount of overshooting or undershooting is allowed by a parameter vector $c \in \mathbb{R}^n$ whose potential is determined by a convex function $g(c)$ that is combined with the entropy term $H(p)$.

In the case of ℓ_2 regularization, the potential function for the primal problem is a quadratic penalty of the form $\frac{1}{2\gamma}\sum_i c_i^2$ for $\gamma = \frac{1}{\sigma_i^2}, i = 1, \ldots, n$ (Lebanon and Lafferty, 2001). In order to recover the specific form of the primal problem for our case, we have to start from the given dual problem. Following Lebanon and Lafferty (2001), the dual function for regularized estimation can be expressed in terms of the dual function $\Lambda(p_{\boldsymbol{\lambda}}, \boldsymbol{\lambda})$ for the unregularized case and the convex conjugate $g^*(\boldsymbol{\lambda})$ of the potential function $g(c)$. In our case the negative of $\Lambda(p_{\boldsymbol{\lambda}}, \boldsymbol{\lambda})$ corresponds to the likelihood term $L(\boldsymbol{\lambda})$, and the negative of the convex conjugate $g^*(\boldsymbol{\lambda})$ is the ℓ_1 regularizer. Thus our dual problem can be stated as

$$\begin{aligned}\boldsymbol{\lambda}^\star &= \arg\max_{\boldsymbol{\lambda}} \Lambda(p_{\boldsymbol{\lambda}}, \boldsymbol{\lambda}) - g^*(\boldsymbol{\lambda}) \\ &= \arg\min_{\boldsymbol{\lambda}} L(\boldsymbol{\lambda}) + \gamma\|\boldsymbol{\lambda}\|_1^1\end{aligned}$$

Since for convex and closed functions, the conjugate of the conjugate is the original function, i.e. $g^{**} = g$ (Boyd and Vandenberghe, 2004), the potential function $g(c)$ for the primal problem can be recovered by calculating the conjugate g^{**} of the conjugate $g^*(\boldsymbol{\lambda}) = \gamma\|\boldsymbol{\lambda}\|_1^1$. In our case, we get

$$g^{**}(c) = g(c) = \begin{cases} 0 & \|c\|_\infty \leq \gamma \\ \infty & \text{otherwise} \end{cases} \qquad (2)$$

where $\|c\|_\infty = \max\{|c_1|, \ldots, |c_n|\}$. A proof for this proposition is given in the Appendix. The resulting potential function $g(c)$ is the indicator function on the interval $[-\gamma, \gamma]$. That is, it restricts the allowable amount of constraint relaxation to at most $\pm\gamma$. From this perspective, increasing γ means to allow for more slack in constraint satisfaction, which in turn allows to fit a more uniform, less overfitting distribution to the data. For features that are included in the model, the parameter values have to be adjusted away from zero to meet the constraints

$$|p[f_i] - \tilde{p}[f_i]| \leq \gamma, \; i = 1, \ldots, n \qquad (3)$$

Initialization: Initialize selected features S to $\emptyset$, and zero-weighted features Z to the full feature set, yielding the uniform distribution $p_{\boldsymbol{\lambda}^{(0)}, S^{(0)}}$.

n-best grafting: For steps $t = 1, \ldots, T$,
(1) for all features f_i in $Z^{(t-1)}$, calculate

$$\left|\frac{\partial L(\boldsymbol{\lambda}^{(t-1)}, S^{(t-1)})}{\partial \lambda_i}\right| > \gamma,$$

(2) $S^{(t)} := S^{(t-1)} \cup N^{(t)}$ and $Z^{(t)} := Z^{(t-1)} \setminus N^{(t)}$ where $N^{(t)}$ is the set of n-best features passing the test in (1),
(3) perform conjugate gradient optimization to find the optimal model $p_{\boldsymbol{\lambda}^\star, S^{(t)}}$ where $\boldsymbol{\lambda}$ is initialized at $\boldsymbol{\lambda}^{(t-1)}$, and $\boldsymbol{\lambda}^{(t)} := \boldsymbol{\lambda}^\star = \arg\max_{\boldsymbol{\lambda}} C(\boldsymbol{\lambda}, S^{(t)})$.

Stopping condition: Stop if for all f_i in $Z^{(t-1)}$:

$$\left|\frac{\partial L(\boldsymbol{\lambda}^{(t-1)}, S^{(t-1)})}{\partial \lambda_i}\right| \leq \gamma$$

Figure 1: n-best gradient feature testing

For features that meet the constraints without parameter adjustment, parameter values can be kept at zero, effectively discarding the features. Note that equality of equations 3 and 1 connects the maximum entropy problem to likelihood regularization.

4 Standardization

Note that the Ω_{p} regularizer presented above penalizes the model parameters uniformly, corresponding to imposing a uniform variance onto all model parameters. This motivates a normalization of input data to the same scale. A standard technique to achieve this is to linearly rescale each feature count to zero mean and standard deviation of one over all training data. The same rescaling has to be done for training and application of the model to unseen data. As we will see in the experimental evaluation presented below, a standardization of input data can also dramatically improve convergence behavior in unregularized optimization . Furthermore, parameter values estimated from standardized feature counts are directly interpretable to humans. Combined with feature selection, interpretable parameter weights are particularly useful for error analysis of the model's feature design.

5 Incremental n-best Feature Selection

The basic idea of the "grafting" (for "gradient feature testing") algorithm presented by (Perkins et al., 2003) is to assume a tendency of ℓ_1 regularization

to produce a large number of zero-valued parameters at the function's optimum, thus to start with all-zero weights, and incrementally add features to the model only if adjusting their parameter weights away from zero sufficiently decreases the optimization criterion. This idea allows for efficient, incremental feature selection, and at the same time avoids numerical problems caused by the discontinuity of the gradient in ℓ_1 regularization. Furthermore, the regularizer is incorporated directly into a criterion for feature selection, based on the observation made above: It only makes sense to add a feature to the model if the regularizer penalty is outweighed by the reduction in negative log-likelihood. Thus features considered for selection have to pass the following test:

$$\left|\frac{\partial L(\boldsymbol{\lambda})}{\partial \lambda_i}\right| > \gamma$$

In the grafting procedure suggested by (Perkins et al., 2003), this gradient test is applied to each feature, and at each step the feature passing the test with maximum magnitude is added to the model. Adding one feature at a time effectively discards noisy and irrelevant features, however, the overhead introduced by grafting can outweigh the gain in efficiency if there is a moderate number of noisy and truly redundant features. In such cases, it is beneficial to add a number of $n > 1$ features at each step, where n is adjusted by cross-validation or on a held-out data set. In the experiments on maximum-entropy parsing presented below, a feature set of linguistically motivated features is used that exhibits only a moderate amount of redundancy. We will see that for such cases, n-best feature selection considerably improves computational complexity, and also achieves slightly better generalization performance.

After adding $n \geq 1$ features to the model in a grafting step, the model is optimized with respect to all parameters corresponding to currently included features. This optimization is done by calling a gradient-based general purpose optimization routine for the regularized objective function. We use a conjugate gradient routine for this purpose (Minka, 2001; Malouf, 2002)[2]. The gradient of our criterion with respect to a parameter λ_i is:

$$\frac{\partial C(\boldsymbol{\lambda})}{\partial \lambda_i} = \frac{1}{m}\sum_{k=1}^{m}\frac{\partial L(\boldsymbol{\lambda})}{\partial \lambda_i} + \gamma \,\mathrm{sign}(\lambda_i)$$

[2]Note that despite gradient feature testing, the parameters for some features can be driven to zero in conjugate gradient optimization of the ℓ_1-regularized objective function. Care has to be taken to catch those features and prune them explicitly to avoid numerical instability.

The sign of λ_i decides if γ is added or subtracted from the gradient for feature f_i. For a feature that is newly added to the model and thus has weight $\lambda_i = 0$, we use the feature gradient test to determine the sign. If $\frac{\partial L(\boldsymbol{\lambda})}{\partial \lambda_i} > \gamma$, we know that $\frac{\partial C(\boldsymbol{\lambda})}{\partial \lambda_i} > 0$, thus we let $\mathrm{sign}(\lambda_i) = -1$ in order to decrease C. Following the same rationale, if $\frac{\partial L(\boldsymbol{\lambda})}{\partial \lambda_i} < -\gamma$ we set $\mathrm{sign}(\lambda_i) = +1$. An outline of an n-best grafting algorithm is given in Fig. 1.

6 Experiments

6.1 Train and Test Data

In the experiments presented in this paper, we evaluate ℓ_2, ℓ_1, and ℓ_0 regularization on the task of stochastic parsing with maximum-entropy models For our experiments, we used a stochastic parsing system for LFG that we trained on section 02-21 of the UPenn Wall Street Journal treebank (Marcus et al., 1993) by discriminative estimation of a conditional maximum-entropy model from partially labeled data (see Riezler et al. (2002)). For estimation and best-parse searching, efficient dynamic-programming techniques over features forests are employed (see Kaplan et al. (2004)). For the setup of discriminative estimation from partially labeled data, we found that a restriction of the training data to sentences with a relatively low ambiguity rate was possible at no loss in accuracy compared to training from all sentences. Furthermore, data were restricted to sentences of which a discriminative learner can possibly take advantage, i.e. sentences where the set of parses assigned to the labeled string is a proper subset of the parses assigned to the unlabeled string. Together with a restriction to examples that could be parsed by the full grammar and did not have to use a backoff mechanism of fragment parses, this resulted in a training set of 10,000 examples with at most 100 parses. Evaluation was done on the PARC 700 dependency bank[3], which is an LFG annotation of 700 examples randomly extracted from section 23 of the UPenn WSJ treebank. To tune regularization parameters, we split the PARC 700 into a heldout and test set of equal size.

6.2 Feature Construction

Table 1 shows the 11 feature templates that were used in our experiments to create $60,109$ features. On the around 300,000 parses for 10,000 sentences in our final training set, $10,986$ features were active, resulting in a matrix of active features times parses that has 66 million non-zero entries. The scale of this experiment is comparable to experiments where

[3]http://www2.parc.com/istl/groups/nltt/fsbank/

Table 1: Feature templates

name	parameters	activation condition
Local Templates		
cs_label	*label*	constituent *label* is present in parse
cs_adj_label	*parent_label*, *child_label*	constituent *child_label* is child of constituent *parent_label*
cs_right_branch		constituent has right child
cs_conj_nonpar	*depth*	non-parallel conjuncts within *depth* levels
fs_attrs	*attrs*	f-structure attribute is one of *attrs*
fs_attr_value	*attr*, *value*	attribute *attr* has value *value*
fs_attr_subsets	*attr*	sum of cardinalities of subsets of *attr*
lex_subcat	*pred*, *args_sets*	verb *pred* has one of *args_sets* as arguments
Non-Local (Top-Down) Templates		
cs_embedded	*label*, *size*	chain of *size* constituents labeled *label* embedded into one another
cs_sub_label	*ancestor_label*, *descendant_label*	constituent *descendant_label* is descendant of *ancestor_label*
fs_aunt_subattr	*aunts*, *parents*, *descendants*	one of *descendants* is descendant of one of *parents* which is a sister of one of *aunts*

much larger, but sparser feature sets are employed[4]. The reason why the matrix of non-zeroes is less sparse in our case is that most of our feature templates are instantiated to linguistically motivated cases, and only a few feature templates encode all possible conjunctions of simple feature tests. Redundant features are introduced mostly by the latter templates, whereas the former features are generalizations over possible combinations of grammar constants. We conjecture that feature sets like this are typical for natural language applications.

Efficient feature detection is achieved by a combination of hashing and dynamic programming on the packed representation of c- and f-structures (Maxwell and Kaplan, 1993). Features can be described as local and non-local, depending on the size of the graph that has to be traversed in their computation. For each local template one of the parameters is selected as a key for hashing. Non-local features are treated as two (or more) local sub-features. Packed structures are traversed depth-first, visiting each node only once. Only the features keyed on the label of the current node are considered for matching. For each non-local feature, the contexts of matching subfeatures are stored at the respective nodes, propagated upward in dynamic programing fashion, and conjoined with contexts of other subfeatures of the feature. Fully matched features are associated with the corresponding contexts resulting in a feature-annotated and/or-forest. This annotated and/or forest is exploited for dynamic programming computation in estimation and best parse selection.

6.3 Experimental Results

Table 2 shows the results of an evaluation of five different systems of the test split of the PARC 700 dependency bank. The presented systems are unregularized maximum-likelihood estimation of a log-linear model including the full feature set (*mle*), standardized maximum-likelihood estimation as described in Sect. 4 (*std*), ℓ_0 regularization using frequency-based cutoff, ℓ_1 regularization using n-best grafting, and ℓ_2 regularization using a Gaussian prior. All ℓ_{p} regularization runs use a standardization of the feature space. Special regularization parameters were adjusted on the heldout split, resulting in a cutoff threshold of 16, and penalization factors of 20 and 100 for ℓ_1 and ℓ_2 regularization respectively, with an optimal choice of 100 features to be added in each n-best grafting step. Performance of these systems is evaluated firstly with respect to F-score on matching dependency relations. Note that the F-score values on the PARC 700 dependency bank range between a lower bound of 68.0% for averaging over all parses and an upper bound of 83.6% for the parses producing the best possible matches. Furthermore, compression of the full feature set by feature selection, number of conjugate gradient iterations, and computation time (in hours:minutes of elapsed time) are reported.[5]

[4] For example, Malouf (2002) reports a matrix of non-zeroes that has 55 million entries for a shallow parsing experiment where 260,000 features were employed.

[5] All experiments were run on one CPU of a dual processor AMD Opteron 244 with 1.8GHz clock speed and 4GB of main memory.

Table 2: F-score, compression, number of iterations, and elapsed time for unregularized and standardized maximum-likelihood estimation, and ℓ_0, ℓ_1, and ℓ_2 regularization on test split of PARC 700 dependency bank.

	mle	*std*	ℓ_0	ℓ_2	ℓ_1
F-score	77.9	78.1	78.1	78.9	79.3
compr.	0	0	18.4	0	82.7
cg its.	761	371	372	34	226
time	129:12	66:41	60:47	6:19	5:25

Unregularized maximum-likelihood estimation using the full feature set exhibits severe overtraining problems, as the relation of F-score to the number of conjugate gradient iterations shows. Standardization of input data can alleviate this problem by improving convergence behavior to half the number of conjugate gradient iterations. ℓ_0 regularization achieves its maximum on the heldout data for a threshold of 16, which results in an estimation run that is slightly faster than standardized estimation using all features, due to a compression of the full feature set by 18%. ℓ_2 regularization benefits from a very tight prior (standard deviation of 0.1 corresponding to penalty 100) that was chosen on the heldout set. Despite the fact that no reduction of the full feature set is achieved, this estimation run increases the F-score to 78.9% and improves computation time by a factor of 20 compared to unregularized estimation using all features. ℓ_1 regularization for n-best grafting, however, even improves upon this result by increasing the F-score to 79.3%, further decreasing computation time to 5:25 hours, at a compression of the full feature set of 83%.

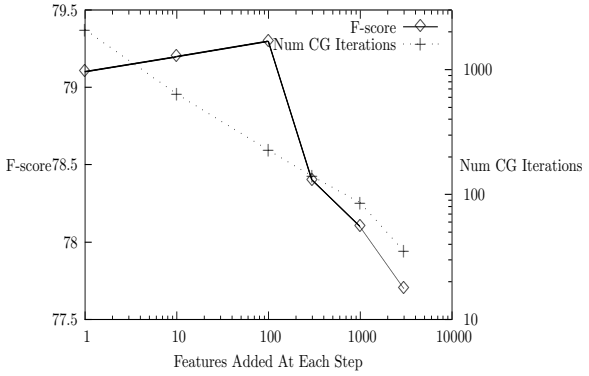

Figure 2: n-best grafting with n of features added at each step plotted against F-score on test set and conjugate gradient iterations.

As shown in Fig. 2, for feature selection from linguistically motivated feature sets with only a moderate amount of truly redundant features, it is crucial to choose the right number n of features to be added in each grafting step. The number of conjugate gradient iterations decreases rapidly in the number of features added at each step, whereas F-score evaluated on the test set does not decrease (or increases slightly) until more than 100 features are added in each step. 100-best grafting thus reduces estimation time by a factor of 10 at no loss in F-score compared to 1-best grafting. Further increasing n results in a significant drop in F-score, while smaller n is computationally expensive, and also shows slight overtraining effects.

Table 3: F-score, compression, number of iterations, and elapsed time for gradient-based incremental feature selection without regularization, and with ℓ_2, and ℓ_1 regularization on test split of PARC 700 dependency bank.

	mle-ifs	ℓ_2-*ifs*	ℓ_1
F-score	78.8	79.1	79.3
compr.	88.1	81.7	82.7
cg its.	310	274	226
time	6:04	6:56	5:25

In another experiment we tried to assess the relative contribution of regularization and incremental feature selection to the ℓ_1-grafting technique. Results of this experiments are shown in Table 3. In this experiment we applied incremental feature selection using the gradient test described above to unregularized maximum-likelihood estimation (*mle-ifs*) and ℓ_2-regularized maximum-likelihood estimation (ℓ_2-*ifs*). Threshold parameters γ are adjusted on the heldout set, in addition to and independent of regularization parameters such as the variance of the Gaussian prior. Results are compared to ℓ_1-regularized grafting as presented above. For all runs a number of 100 features to be added in each grafting step is chosen. The best result for the *mle-ifs* run is achieved at a threshold of 25, yielding an F-score of 78.8%. This shows that incremental feature selection is a powerful tool to avoid overfitting. A further improvement in F-score to 79.1% is achieved by combining incremental feature selection with the ℓ_2 regularizer at a variance of 0.1 for the Gaussian prior and a threshold of 15. Both runs provide excellent compression rates and convergence times. However, they are still outperformed by the ℓ_1 run that achieves a slight improvement in F-score to 79.3% and a slightly better runtime. Furthermore, by integrating regularization naturally into thresh-

olding for feature selection, a separate thresholding parameter is avoided in ℓ_1-based incremental feature selection.

A theoretical account of the savings in computational complexity that can be achieved by n-best grafting can be given as follows. Perkins et al. (2003) assess the computational complexity for standard gradient-based optimization with the full feature set by $\approx cmp^2\tau$, for a multiple c of p line minimizations for p derivatives over m data points, each of which has cost τ. In contrast, for grafting, the cost is assessed by adding up the costs for feature testing and optimization for s grafting steps as $\approx (msp+\frac{1}{3}cms^3)\tau$. For n-best grafting as proposed in this paper, the number of steps can be decomposed into $s = n \cdot t$ for n features added at each of t steps. This results in a cost of $\approx mtp$ for feature testing, and $\approx \frac{1}{3}cmn^2t^3\tau$ for optimization. If we assume that $t \ll n \ll s$, this indicates considerable savings compared to both 1-best grafting and standard gradient-based optimization.

7 Discussion and Conclusion

A related approach to ℓ_1 regularization and constraint-relaxation for maximum-entropy modeling has been presented by Kazama and Tsujii (2003). In this approach, constraint relaxation is done by allowing two-sided inequality constraints

$$-B_i \leq \tilde{p}[f_i] - p[f_i] \leq A_i,\ A_i, B_i > 0$$

in entropy maximization. The dual function is the regularized likelihood function

$$\frac{1}{m}\sum_{j=1}^{m} p_{\boldsymbol{\alpha}-\boldsymbol{\beta}}(x_j|y_j) - \sum_{i=1}^{n}\alpha_i A_i - \sum_{i=1}^{n}\beta_i B_i$$

where the two parameter vectors $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ replace our parameter vector $\boldsymbol{\lambda}$, and $\alpha_i, \beta_i \geq 0$. This regularizer corresponds to a simplification of double-sided exponentials to a one-sided exponential distribution which is non-zero only for non-negative parameters. The use of one-sided exponential priors for log-linear models has also been proposed by Goodman (2003), however, without a motivation in a maximum entropy framework. The fact that Kazama and Tsujii (2003) allow for lower and upper bounds of different size requires the parameter space to be doubled in their approach. Furthermore, similar to Goodman (2003), the requirement to work with a one-sided strictly positive exponential distribution makes it necessary to double the feature space to account for (dis)preferences in terms of strictly positive parameter values. These are considerable computational and implementational disadvantages of these approaches. More importantly, an integration of ℓ_1 regularization into incremental feature selection was not considered.

Incremental feature selection has been proposed firstly by Della Pietra et al. (1997) in a likelihood-based framework. In this approach, an approximate gain in likelihood for adding a feature to the model is used as feature selection criterion, and thresholds on this gain are used as stopping criterion. Maximization of approximate likelihood gains and gradient feature testing both are greedy approximations to the true gain in the objective function - grafting can be seen as applying one iteration of Newton's method, where the weight of the newly added feature is initialized at 0, to calculate the approximate likelihood gain. Efficiency and accuracy of both approaches are comparable, however, the grafting framework provides a well-defined mathematical basis for feature selection and optimization by incorporating selection thresholds naturally as penalty factors of the regularizer. The idea of adding n-best features in each selection step also has been investigated earlier in the likelihood-based framework (see for example McCallum (2003)). However, the possible improvements in computational complexity and generalization performance due to n-best selection were not addressed explicitly. Further improvements of efficiency of grafting are possible by applying Zhou et al.'s (2003) technique of restricting feature selection in each step to the top-ranked features from previous stages.

In sum, we presented an application of ℓ_1 regularization to likelihood maximization for log-linear models that has a simple interpretation as bounded constraint relaxation in terms of maximum entropy estimation. The presented n-best grafting method does not require specialized algorithms or simplifications of the prior, but allows for an efficient, mathematically well-defined combination of feature selection and regularization. In an experimental evaluation, we showed n-best grafting to outperform ℓ_0, 1-best ℓ_1, ℓ_2 regularization and standard incremental feature selection in terms of computational efficiency and generalization performance.

References

Stephen Boyd and Lieven Vandenberghe. 2004. *Convex Optimization*. Cambridge University Press.

Stanley F. Chen and Ronald Rosenfeld. 1999. A gaussian prior for smoothing maximum entropy models. Technical Report CMU-CS-99-108, Carnegie Mellon University, Pittsburgh, PA.

Stephen Della Pietra, Vincent Della Pietra, and John Lafferty. 1997. Inducing features of random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(4):380–393.

Joshua Goodman. 2003. Exponential priors for maximum entropy models. Unpublished Manuscript, Microsoft Research, Redmont, WA.

Mark Johnson, Stuart Geman, Stephen Canon, Zhiyi Chi, and Stefan Riezler. 1999. Estimators for stochastic "unification-based" grammars. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL'99)*, College Park, MD.

Ronald M. Kaplan, Stefan Riezler, Tracy H. King, John T. Maxwell III, and Alexander Vasserman. 2004. Speed and accuracy in shallow and deep stochastic parsing. In *Proceedings of the Human Language Technology conference / North American chapter of the Association for Computational Linguistics annual meeting (HLT/NAACL'04)*, Boston, MA.

Jun'ichi Kazama and Jun'ichi Tsujii. 2003. Evaluation and extension of maximum entropy models with inequality constraints. In *Proceedings of EMNLP'03*, Sapporo, Japan.

Guy Lebanon and John Lafferty. 2001. Boosting and maximum likelihood for exponential models. In *Advances in Neural Information Processing 14 (NIPS'01)*, Vancouver.

Robert Malouf. 2002. A comparison of algorithms for maximum entropy parameter estimation. In *Proceedings of Computational Natural Language Learning (CoNLL'02)*, Taipei, Taiwan.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of english: The Penn treebank. *Computational Linguistics*, 19(2):313–330.

John Maxwell and Ron Kaplan. 1993. The interface between phrasal and functional constraints. *Computational Linguistics*, 19(4):571–589.

Andrew McCallum. 2003. Efficiently inducing features of conditional random fields. In *Proceedings of the 19th Conference on Uncertainty in Artificial Intelligence (UAI'03)*, Acapulco, Mexico.

Thomas Minka. 2001. Algorithms for maximum-likelihood logistic regression. Department of Statistics, Carnegie Mellon University.

Andrew Y. Ng. 2004. Feature selection, l_1 vs. l_2 regularization, and rotational invariance. In *Proceedings of the 21st International Conference on Machine Learning*, Banff, Canada.

Simon Perkins, Kevin Lacker, and James Theiler. 2003. Grafting: Fast, incremetal feature selection by gradient descent in function space. *Machine Learning*, 3:1333–1356.

Stefan Riezler, Tracy H. King, Ronald M. Kaplan, Richard Crouch, John T. Maxwell, and Mark Johnson. 2002. Parsing the Wall Street Journal using a Lexical-Functional Grammar and discriminative estimation techniques. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL'02)*, Philadelphia, PA.

Robert Tibshirani. 1996. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B*, 58(1):267–288.

Yaqian Zhou, Fuliang Weng, Lide Wu, and Hauke Schmidt. 2003. A fast algorithm for feature selection in conditional maximum entropy modeling. In *Proceedings of EMNLP'03*, Sapporo, Japan.

Appendix: Proof of Proposition 2

Following Boyd and Vandenberghe (2004), the convex conjugate of function $g : \mathbb{R}^n \rightarrow \mathbb{R}$ is

$$g^*(w) = \sup_u \{\sum_{i=1}^n w_i u_i - g(u)\}$$

and the dual norm $\|\cdot\|_*$ of norm $\|\cdot\|$ on $\mathbb{R}^n$ is

$$\|w\|_* = \sup_u \{\sum_{i=1}^n w_i u_i| \ \|u\| \leq 1\} \qquad (4)$$

and the dual norm of the ℓ_1 norm is the ℓ_∞ norm

$$\|w\|_* = \|w\|_\infty \text{ for } \|u\| = \|u\|_1^1 \qquad (5)$$

We show that the convex conjugate of

$$g(u) = \gamma\|u\|_1^1, \text{ for } \gamma > 0$$

$$\text{is } g^*(w) = \begin{cases} 0 & \|w\|_\infty \leq \gamma \\ \infty & \text{otherwise} \end{cases}$$

Proof. Let $\|w\|_\infty \leq \gamma$, then $\sum_i w_i u_i \leq \|u\|_1^1 \|w\|_\infty$ (from 4 and 5) $\leq \|u\|_1^1 \gamma$ (since $\|w\|_\infty \leq \gamma$). Then $\sum_i w_i u_i - \|u\|_1^1 \gamma \leq 0$ and $u = 0$ maximizes it with maximum value $g^*(w) = 0$.

Let $\|w\|_\infty > \gamma$, then $\exists z$ s.t. $\|z\|_1^1 \leq 1$ and $\sum_i w_i z_i > \gamma$ (from 4 and 5). For $u = tz$, let $t \rightarrow \infty$, then $\sum_i w_i u_i - \gamma\|u\|_1^1 = t(\sum_i w_i z_i - \gamma\|z\|_1^1) \rightarrow \infty$ (since $\sum_i w_i z_i - \gamma\|z\|_1^1 > 0$), thus $g^*(w) = \infty$.