Recursive Feature Elimination for Multi-Class Genre Identification

Zhili Wu

Serge Sharoff

Katja Markert

Centre for Translation Studies
University of Leeds
Leeds, LS2 9JT, UK

ABSTRACT

Recent research in automatic genre identification shows that some of the best performing features are based on character n-grams. However, the amount of character n-grams extracted from a modest training corpus is often large. It is necessary to perform extensive feature selection to achieve compact machine learning models, which can also keep comparable or better classification performance.

The present paper extends binary-class Recursive Feature Elimination (RFE) to a range of multi-class L2/L1 regularized methods, including typical L_2 regularized SVMs with different multi-class decomposition, the integrated multi-class SVM proposed by Crammer and Singer, and the L_1 regularized SVM that leads to zero weights to features, and additionally L_2 and L_1 regularized logistic regression. Results show that all SVM variants can benefit from the multi-class RFE to reduce the number of features while keeping or improving classification performance. L_1 regularized methods benefit less from RFE due to its own mechanism of pruning features without needing further recursions for feature elimination.

Categories and Subject Descriptors

[**Machine Learning**]: classification

General Terms

Keywords

Genre classification, Feature Selection, SVM, L_2 Regularization, L_1 Regularization

1. INTRODUCTION

The explosive growth of the Web and other document collections results in the need to classify texts into categories other than topics to improve the performance of search engines, targeted crawling, document routing and other applications. However, Automatic Genre Identification (AGI) needs features beyond conventional term frequencies. The genre of a text is often independent from its topic, e.g., newswires vs blogs vs research articles can be on a variety of topics, but still in the same genre. Therefore, AGI should be robust with respect to the topics of documents, as well as to their format (HTML, PDF, MS Word).

Recent research [3, 9] identified character n-grams extracted from plain text as one of the best performing features over a range of document collections. In addition to their good performance, it is very easy to extract them, they do not need any language-dependent tools, such as stemmers, lemmatizers, part-of-speech taggers or parsers. Their performance is related to their ability to generalise over a range of document properties, including endings (e.g., Latin-derived words are more common in research aritcles, dates are more common in newswires), punctuation (question marks are more common in FAQs), as well as genre-specific words (*however* is more common in discussions). However, there are considerably more character n-grams in a given corpus in comparison to keywords used in topic detection, so we need to perform a more aggressive feature selection in AGI.

This report summarizes genre classification experiments over several genre collections, and extends binary-class Recursive Feature Elimination (RFE) to a range of multi-class L2/L1 regularized methods. Results show that all SVM variants can benefit from the multi-class RFE for reducing the needed features while keeping/improving classification performance. L_1 regularized methods benefit less from RFE due to its own mechanism of pruning many features without needing further recursions of feature elimination.

2. THEORETICAL SETUP

2.1 Definition of Multiclass SVMs

A binary linear SVM assigns weights to features and outputs the prediction through a linear function:

$$f(\mathbf{x}) = \sum_{i=1}^{d} w_i x_i + b, \tag{1}$$

where $\mathbf{x}$ is a document represented as a $d-$dimensional feature vector, w_i is a weight, and b is a scalar intercept.

$\mathbf{w}$ and b are determined by optimization of a regularized objective function H

$$\min \ H(\mathbf{w}, b): \quad Regularizer(\mathbf{w}) + CLoss(\mathbf{y}, \mathbf{w}^T\mathbf{x} + b),$$

where C is a user specified parameter. Table 1 lists the specific regularizer and loss functions for all of the seven

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SIGIR '2010, Feature Selection Workshop, Geneva

Copyright 200X ACM X-XXXXX-XX-X/XX/XX ...$10.00.

Table 1: Regularized Primal Formulations of SVM and Logistic Regression

Abbreviation	Regularizer	Loss	Multi-class	Package Used
L2L1-OVOSVM	$\mathbf{w}^T\mathbf{w}$	$\sum_{i=1}^{l}\max(0, 1 - y_i(\mathbf{w}^T\mathbf{x}_i + b))$	One-Vs-One	Libsvm (-t 0)
L2L1-SVM	$\mathbf{w}^T\mathbf{w}$	$\sum_{i=1}^{l}\max(0, 1 - y_i\mathbf{w}^T\mathbf{x}_i)$	One-Vs-All	LibLinear (-s 3)
L2L2-SVM	$\mathbf{w}^T\mathbf{w}$	$\sum_{i=1}^{l}(1 - y_i\mathbf{w}^T\mathbf{x}_i)^2$	One-Vs-All	LibLinear (-s 1)
L1L2-SVM	$\lvert\lvert\mathbf{w}\rvert\rvert_1$	$\sum_{i=1}^{l}(1 - y_i\mathbf{w}^T\mathbf{x}_i)^2$	One-Vs-All	LibLinear (-s 5)
L1-LR	$\lvert\lvert\mathbf{w}\rvert\rvert_1$	$\sum_{i=1}^{l} log(1 + e^{-y_i\mathbf{w}^T\mathbf{x}_i})$	One-Vs-All	LibLinear (-s 6)
L2-LR	$\mathbf{w}^T\mathbf{w}$	$\sum_{i=1}^{l} log(1 + e^{-y_i\mathbf{w}^T\mathbf{x}_i})$	One-Vs-All	LibLinear (-s 0)
L2L1-CSSVM	$\sum_{m=1}^{k}\mathbf{w}_m^T\mathbf{w}_m$ k:number of classes	$\sum_{i=1}^{l}\max_m\{h(y_i, m) - \mathbf{w}_{y_i}^T\mathbf{x}_i + \mathbf{w}_m^T\mathbf{x}_i\}$ $h(y_i, m) = 1,\ \forall y_i \neq m; h(y_i, y_i) = 0$	Integrated	LibLinear (-s 4)

methods tested in the present paper. A name L2L1-SVM denotes SVM with L_2 norm regularization and L_1 loss. L2L1-CSSVM denotes the Crammer-Singer-SVM, which relies on an integrated objective function for multi-class tasks. Other methods deal with decomposed binary subproblems. L2L1-OVOSVM relies on the One-Vs-One decomposition to classify a class pair in each subproblem; and others relies on One-Vs-All to classify a class from all other classes. We use the L2L1-OVOSVM implemented in LibSVM, and others in LibLinear. The methods used in LibLinear append b to $\mathbf{w}$. Therefore, there is no separate b in their objective functions.

2.2 Feature Selection with Binary SVMs

It is intuitive to eliminate features that are associated with smaller weights in the decision function (1). For binary tasks the L2L1-OVOSVM reduces to an ordinary binary SVM. As shown in [2] and [10] pruning a feature with the smallest weight in a binary SVM, will cause the smallest change in the objective function from the current optimum, due to the following relation:

$$\Delta H(\text{without feature } q) \approx \frac{1}{2}\frac{\partial^2 H}{\partial w_q^2}w_q^2 = w_q^2. \qquad (2)$$

In practice, the process of pruning features with small magnitude can be recursively applied. And in each iteration more than one feature can be removed. Since the model may not be optimal after feature removal, a new model is then built after each iteration.

2.3 Feature Selection with Multiclass SVMs

In the following we discuss different RFE extensions for different multiclass SVMs from Table 1.

One-Vs-One SVM.

The one-vs-one approach for a $k-$class problem will generate $k(k-1)/2$ binary SVMs. As a straightforward option we can apply RFE techniques to each binary SVM, and then pool all these features specific to each SVM together. However, this approach will generate a large feature set during the feature combination stage. Moreover, treating each binary SVM separately needs us to tune it separately, which will cause high complexity.

RFE for binary SVMs adopts a single weight for each feature. The same principle can be borrowed for one-vs-one SVM. If we have a k-class problem and want to remove the $q-$th feature simultaneously for all SVM components, we can define the following squared quantity v_k^2, the summation of all the squared weights for the $q-$th feature obtained from all SVMs:

$$v_q^2 = \sum_{1 \leq i < j \leq k} w(ij)_q^2. \qquad (3)$$

As a justification, removing a feature with the smallest v_k^2, will minimally alter the overall objective function of all pairwise SVMs, assuming each contributes equally to the overall multiclass SVM:

$$\begin{aligned} \Delta H(\text{no feature } q) &= \frac{2}{k(k-1)}\sum_{1 \leq i < j \leq k}\Delta H^{(ij)}(\text{no feature } q) \\ &\approx \frac{2}{k(k-1)}\sum_{1 \leq i < j \leq k}\frac{1}{2}\frac{\partial^2 H^{(ij)}}{\partial w(ij)_q^2}w(ij)_q^2 \\ &= \frac{2}{k(k-1)}\sum_{1 \leq i < j \leq k} w(ij)_q^2 = v_q^2, \qquad (4) \end{aligned}$$

L_2 Regularized LR and One-Vs-All/Integrated SVMs.

For L_2 regularized methods that rely on one-vs-all decomposition, or integrated formulation, k weight vectors will be obtained, one per class. Similar to the summation of squared weights for one-vs-one SVM, a general RFE criterion can be proposed for one-vs-all SVMs as the following:

$$v_q^2 = \sum_{i=1}^{k} w(i)_q^2. \qquad (5)$$

For one-vs-all L_2 regularized SVMs, this measure has been given in [4] for their *simultaneous multi-class feature selection.* Note their theoretical justification for the squared quantity is derived from a framework that introduces scaling factors and relies on approximation techniques. In [10], the same quantity has been developed for both the one-vs-all multiclass SVM and the Crammer-Singer-SVM.

L_1 Regularized SVM/LR.

For L_1 Regularized L1L2-SVM and L1-LR, the analysis in 2 does not hold. However, a nice property of L_1 regularized methods is that they often lead to zero weights, thus achieving automatic feature pruning. However, when the desired number of features is small, the optimal model may not generate many zero weights, thus necessitating the pruning of features with non-zero weights. Nevertheless, the summed-up squared quantity can still be used as an approximate measure for RFE. It is also possible to use the summation of all absolute weights for each feature dimension, $\sum_{1 \leq i \leq k}|w(i)_q|$, or similar to the quantity in [1], the maximum of these absolute weights $\max_{1 \leq i \leq k}|w(i)_q|$.

Table 2: Genre corpora used

Source	# texts	# genres	# features
I-EN-Sample [8]	250	7	81423
KI-04 [6]	1205	8	270185
SANTINIS [7]	1400	7	334044
Brown Corpus [5]	500	10	76508

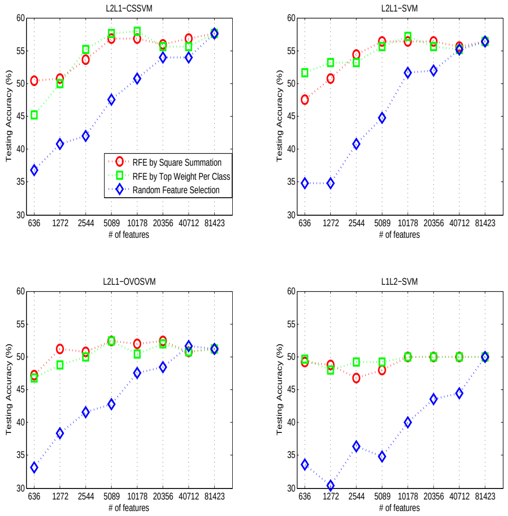

Figure 1: RFE vs Random Feature Selection in I-EN

3. EXPERIMENTAL SETUP

Several collections of webpages are used for AGI research. Table 2 lists the corpora used in this experiment. In addition to webgenres proper we also experimented with the classic Brown Corpus, since it has a diverse set of genres. For features we used character tetragrams tested in two variants: *TFIDF*-selected features vs. using *binary* features.

We used RFE based on squared summation as well as a new RFE strategy, in which we ranked each weight vector, and selected a feature with the largest weight in a round-robin manner till a desired number of features is obtained (referred to as 'RFE by Top Weight per Class').

In our experiment, we carry out 10-fold cross validation based on non-stratified random sampling. In each fold we have 10% data for testing, 80% data for training, leaving the 10% as a validation set to tune the parameter C. For the 80% data, we first use all features to train a classifier that has the best validation performance. Then we apply RFE pruning and get a subset of features, which are used to update the training, testing and validation sets for next round of feature selection.

4. RESULTS

Figure 1 shows the testing accuracy curves of two RFE approaches and the baseline random feature selection on the I-EN corpus. As half of the features are pruned iteration by iteration, the testing accuracy achieved by random selection drops quickly. However, even with only 2544 features, accounting for 1/32 of the whole feature set, the accuracy of RFE approaches haven't shown large decreases. For L2L1-CSSVM and L2L1-SVM, their curves even show a slight increase within the feature range from 5089 to 10178. Note due to the capability of SVM in dealing with high-dimensional sparse data, feature pruning may not necessarily bring significant accuracy gain. However, feature pruning can bring significant speedup for the final training model and the prediction phase. Since the theoretic complexity of linear SVMs is linearly proportional to the feature dimension, by pruning the features from 81423 to 5089, a ten-fold speedup in final training and testing is observed in our experiment.

In a smaller experiment, we focus on how the 598 features perform on the four of ten Brown genres: religion, gov. doc, learned and fiction. The L2L1-SVM based on the 598 features has a 10-fold cross validation score 94.9% on the four classes of data, higher than the score 87.7% based on all features and 76.7% based on random features. It is interesting to note the rank of the 589 features by L2L1-SVM weights has no tendency to correlate the rank of these features by their total frequency or document frequency, because the *Spearman rank correlation scores* between the L2L1-SVM rank and the latter two close to zero (-0.0196 and -0.0507, respectively). That means features picked by RFE are quite different from those picked by simply counting frequencies.

RFE with Different Methods and Features.

Table 3 shows the testing accuracy of different methods with RFE. These results on testing sets are based on the number of features that optimize the validation sets. So they are not necessarily the peaks of their testing accuracy curves (see Figure 1). Only the RFE results based on squared summation is used since there are no considerable differences in RFE based on top weights.

For each column of Table 3, the *uparrow* symbol after a value means it is significantly better than any value with a *downarrow*, supported by the 5% one-tail Mcnemar test. It shows that the L2-regularized methods except the L2L1-OVOSVM perform slightly better than L1-regularized ones. This may be explained by the tendency of L1-regularized methods to use less features since they progressively prune features with zero weights (e.g. the long flat end of L1L2-SVM and L1-LR curves in Figure 1 is due to the removal of features with zero weights). However, L1-regularized methods perform well on Santinis and KI-04, which contain clearly separate genre examples. The L1-regularized methods work better for less noisy data.

RFE vs SVD-Based Dimensionality Reduction.

It has been observed L1-regularized methods seem to be more noise-sensitive. To assess this and to compare RFE and typical dimensionality and noise reduction methods, we applied SVD and trained the model on the left singular vectors (they form a $l \times l$ matrix since $l \ll d$). Since each dimension of the singular vectors are naturally ordered by singular values, we just need to prune features sequentially.

For Brown corpus after doing SVD on binary features, the L1-LR and L1L2-SVM enhance their accuracy to 68.4% and 67.2%, respectively, outperforming the third best 66.4% by L2L2-SVM. For I-EN corpus, the L1-LR and L1L2-SVM using SVD features improve to 55.2% and 52%, much closer to the best 55.6% by L2L1-CSSVM using a binary feature subset.

However, for Santinis the accuracy of L1-LR and L1L2-SVM using SVD features decreases to 95.43% and 95.71%, respectively. For KI-04, the accuracy of L1-LR and L1L2-SVM using SVD features decreases to 80.86% and 80.51%,

Table 3: RFE with Different Methods and Features

Method	I-EN		BROWN		SANTINIS		KI-04		(#wins,#loses)
	binary	TFIDF	binary	TFIDF	binary	TFIDF	binary	TFIDF	
L2L1-OVOSVM	52.40↓	51.20	66.20	64.40↓	96.36	93.07↓	82.82↓	78.02↓	**(0,-5)**
L2L1-SVM	56.40↑	51.60	65.60	65.00↓	96.36	94.79	85.48↑	79.10↓	(2,-2)
L2L2-SVM	55.60	52.00	66.80↑	67.00↑	96.21	95.07↑	85.48↑	80.78↑	**(4,0)**
L1L2-SVM	50.00↓	52.80	55.00↓	60.00↓	97.00	94.07↓	82.57↓	80.35	**(0,-5)**
L1-LR	49.20↓	50.00	59.00↓	59.60↓	97.00	95.29↑	83.24↓	81.42↑	(2,-4)
L2-LR	55.20	54.00	64.40↓	65.00	96.43	95.43↑	85.40↑	81.68↑	(3,-1)
L2L1-CSSVM	56.80↑	52.80	66.20↑	66.20	96.71	93.50↓	85.15↑	78.44↓	(3,-2)

# features	**Genre label**	**Top ranking features**
76508	Press/Reportage:	rday &_,_ week sday _wee rida nnou _ann noun e_19 quot;._ frid _th iday esda
599	Press/Reportage:	rday &_,_ rida nnou sday ford _ann quot;._ e_19 ank_ week _tea univ _wee _th onda
76508	Learned (research):	ms_t erab s_by ly_l eris ed_e ompa ly_u naly onta tiga ampl ch_w m_a_ n_ap y_lo alys
599	Learned (research):	ms_t y_li ly_l ._si s_ve jor_ by_c te_c _sol ot_a ompa ed_e sly_ eris ollo plie naly alys

Explanations of features:.
rday=yesterday,Saturday; &=marks abbreviations, e.g., Mr., Co., Okla.; *sday*=Tuesday, Wednesday, Thursday; *_wee*=(last, this) week; *rida*=Friday; *nnou*=announced; *ms_t*=aims/claims/seems/problems/systems to/that; *erab*=considerably; *s_by*= presentations/lines/facilities/methods/beams by; *ly_l,ly_u,sly*=adverbs; *eris*=characteristic/characterized; *ompa*= compare /comparison; *onta*=contain; *tiga*=investigate/investigation; *ampl*=example/sample; *naly, alys*=analysis

respectively. That shows SVD may help less for clean data.

In Table 3 we also present qualitative results of feature selection for L2L2 feature selection on the Brown corpus. Reasonable features selected from the full SVM model (dates and places in newswires, research-related verbs, adverbs and passives in research articles) are mostly retained even after agressive pruning (from 76508 down to 599 features).

5. CONCLUSIONS

This report summarizes genre classification experiments over a selection of different genre collections, and extends binary-class Recursive Feature Elimination (RFE) to a range of multi-class L2/L1 regularized methods. Results show that all SVM variants can benefit from the multi-class RFE for reducing the needed features while keeping/improving classification performance. L_1 regularized methods benefit less from RFE due to its own mechanism of pruning many features without needing further recursions of feature elimination.

As for future work, we are interested in how the features selected by best methods for a corpus can be applied to other corpus if their genre labels are made to be unified.

6. REFERENCES

[1] X.-w. Chen, X. Zeng, and D. van Alphen. Multi-class feature selection for texture classification. *Pattern Recogn. Lett.*, 27(14):1685–1691, 2006.

[2] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik. Gene selection for cancer classification using support vector machines. *Mach. Learn.*, 46(1-3):389–422, 2002.

[3] I. Kanaris and E. Stamatatos. Learning to recognize webpage genres. *Information Processing and Management*, 45:499–512, 2009.

[4] O. C. S. S. Keerthi. Multi-class feature selection with support vector machines. In *Proceedings of the American Statistical Association*, 2008.

[5] H. Kučera and W. N. Francis. *Computational analysis of present-day American English.* Brown University Press, Providence, 1967.

[6] S. Meyer zu Eissen and B. Stein. Genre classification of web pages. In *Proceedings of the 27th German Conference on Artificial Intelligence*, Ulm, Germany, 2004.

[7] M. Santini. Cross-testing a genre classification model for the web. In A. Mehler, S. Sharoff, and M. Santini, editors, *Genres on the Web: Computational Models and Empirical Studies.* Springer, Berlin/New York, 2010.

[8] S. Sharoff. In the garden and in the jungle: Comparing genres in the BNC and Internet. In A. Mehler, S. Sharoff, and M. Santini, editors, *Genres on the Web: Computational Models and Empirical Studies.* Springer, Berlin/New York, 2010.

[9] S. Sharoff, Z. Wu, and K. Markert. The web library of babel: evaluating genre collections. In *Proc. of the Seventh Language Resources and Evaluation Conference, LREC 2010*, Malta, 2010.

[10] X. Zhou and D. P. Tuck. MSVM-RFE. *Bioinformatics*, 23(9):1106–1114, 2007.