A Grouped Structure-based Regularized Regression Model for Text Categorization

Wenbin Zheng[1,2], Yuntao Qian[1⋆], Minchao Ye[1]
[1]College of Computer Science and Technology, Zhejiang University, Hangzhou, China
[2]College of Information Engineering, China Jiliang University, Hangzhou, China
Email: zwb@zju.edu.cn, zjucsai@gmail.com, yeminchao@126.com

Abstract— **The lasso regularization has successfully been used in regression models for feature selection; however, lasso considers all variable to be independent and non-correlative, which will yield an excessively sparse solution (i.e., some important discriminating features might be discarded) if the features are highly correlated. This paper proposes a novel approach in which a sparse model was developed for text categorization. We firstly constructed a grouped structure according the correlation of text features, and then embedded the structure into a regression model via a between- and within- group sparse manner. The goal of such manner is that the groups containing many discriminating features can be selected even the features in these groups are highly correlated, and the noise within the selected groups could be discarded simultaneously, which is beneficial for classification. The experimental results show that the proposed method achieves a good tradeoff between performance and sparsity on three benchmark data sets.**

Index Terms— **Text Categorization, Regularization, Sparse, Lasso, Grouped Structure**

I. Introduction

In text categorization, a major difficulty is the high dimensionality of the input feature space [1]. Feature selection is an effective method for dimensionality reduce [2], [3]. Commonly, features may be selected via a scoring metric, which is the so-called filter approach. On the other hand, the wrapper method [4] searches feature subsets using the classifier itself as part of their function evaluation, which guarantees the consistency between the feature selection and classification.

As a wrapper method to yield sparse models, regularized regression models have received much attention over the past few years. Lasso regularization [5] was originally proposed for linear regression models, and subsequently adapted to the logistic case [6], [7]. It has been shown that the lasso penalization can not only deal with continuous shrinkage, but also has the property of performing automatic variable selection simultaneously.

Nevertheless, the lasso considers all variable to be independent and non-correlative, if there is a group of variables among which the pairwise correlations are relatively high, it tends to select only one variable from the group [8], [9]. Therefore, some useful discriminative features might be ignored in the case, which commonly exists in text categorization because of the nature of languages.

In this paper, we propose a novel approach to deal with this issue. With a clustering algorithm according the correlation of text features, a grouped structure was constructed. And then, the structure was embedded into a logistic regression model via a between- and within-group sparse manner. Thus, the groups that contain many important features can be selected even the features in these groups are highly correlated, and noise within the selected groups could be discarded simultaneously during the model fitting. After that, classification was implemented in this model.

A key of our approach is the combination between the grouped structure regularization and the feature clustering, which could exploit some structured information of text features to improve the performance of the sparse model. Such approach, to our knowledge, has not been done in text categorization.

The rest of this paper is organized as follows. Section II introduces the related work briefly. Section III presents the proposed model, and its solution is given in Section IV. Experimental results and analysis are shown in Section V. Finally, we summarize the conclusions in Section VI.

II. Related Works

The lasso [5], which was originally proposed for linear regression, has become a powerful method for model selection and shrinkage estimation. In addition, Routh [6] extended it to a more generalized case, e.g., the logistic regression model, which has been regarded as one of the best model in text categorization [7], [9]–[11].

Nevertheless, recent researches show that the performance of the lasso is dominated by the ridge when the data sets is short, fat and the correlations among features are relatively high [8], [9]. To tackle the limitations, Zou and Hastie [8] proposed the Elastic net method which tries to capture the best of both L_1 and L_2 penalizations. However, such formulation relies on a particular form of the least square term, which is difficult to be extended to the logistic regression case.

As an alternative model selection and estimation method, Yuan and Lin [12] proposed a novel approach,

This work was supported by the National Nature Science Foundation of China (No.61171151) and the Natural Science Foundation of Zhejiang Province (No.Y6110147 & No. Y1101359).

doi:10.4304/jsw.7.9.2119-2124

called group lasso, in which a technique of selecting the features in a grouped manner was implemented. And then, this method was extended to logistic regression case by Meier et al. [13]. Recently, Friedman [14] proposed a linear regression model, in which the features was selected via a between- and within- group sparse manner. These models have been successfully employed in numerous applications, such as gene finding [13], [15] and the classification of hyperspectral remote sensing image [16].

Motivated by [7] and [12], this paper implemented a logistic regression model for text categorization. Different from [7], our method used a grouped structure-based regression model to select features, such that the important discriminating features could be selected even the correlations among features are relatively high. Different from [12] and [13], our method was implemented in a logistic regression model with a between- and within-group sparse manner.

III. The Proposed Model

Generally, a regularized regression model can be formulated as a penalized optimization problem:

$$\min_{\boldsymbol{\beta}} l(\boldsymbol{\beta}) + \lambda\phi(\boldsymbol{\beta}), \tag{1}$$

where $\boldsymbol{\beta}$ is the parameter vector (regression coefficient), $l(\boldsymbol{\beta})$ is the empirical loss function (e.g., the least squares loss and the logistic loss), $\lambda > 0$ is the regularized parameter, and $\phi(\boldsymbol{\beta})$ is the penalty function.

Concretely, when $\phi(\boldsymbol{\beta}) = ||\boldsymbol{\beta}||_1$ (i.e., lasso), we can obtain a sparse solution. Nevertheless, as discussed above, the solution might ignore many discriminative features if the features are highly correlated. Therefore, we desire that all features containing important discriminative information will be remained in an entire group manner. For this purpose, we divide all features into some groups such that the features in each group are highly correlated. Following [12], we use the grouped structure as the penalty term, which is:

$$\phi(\boldsymbol{\beta}) = \sum_{k=1}^{G} (\sqrt{p_k} \cdot \|\boldsymbol{\beta}_k\|_2), \tag{2}$$

where G is the number of groups, p_k is the number of features in the k-th group, and $\boldsymbol{\beta}_k \in \mathbb{R}^{p_k}$ is the parameter vector corresponding to the k-th group. As per the discussion, such grouped structure regularization can be called group lasso and it encourages sparsity only at the factor level, i.e., it selects or discards features only in a grouped manner.

In this study, we selected logistic regression as the regression model, it can provide a probability value rather than a score, which may be useful in some practical applications. Suppose that we have a training set: $(\mathbf{x}_i, y_i),\ i = 1, \ldots, n$, where $\mathbf{x}_i \in \mathbb{R}^p$ is a p-dimensional vector, $y_i \in \{1, -1\}$ is the corresponding class label encoding membership (1) or nonmembership (-1) of the training document in the category.

We rewrite $\mathbf{x}_i = (\mathbf{x}_{i,1}^{\mathrm{T}}, \mathbf{x}_{i,2}^{\mathrm{T}}, \ldots, \mathbf{x}_{i,G}^{\mathrm{T}})^{\mathrm{T}}$ with a group of vectors $\mathbf{x}_{i,k} \in \mathbb{R}^{p_k},\ k = 1, \ldots, G$. The logistic regression models the conditional probability by:

$$\mathbb{P}_{\boldsymbol{\beta}}(y_i = +1|\mathbf{x}_i) = \frac{1}{1 + \exp(-\eta_{\boldsymbol{\beta}}(\mathbf{x}_i))}, \tag{3}$$

where $\eta_{\boldsymbol{\beta}}(\mathbf{x}_i)$ is the logit function, which is defined by:

$$\eta_{\boldsymbol{\beta}}(\mathbf{x}_i) = \beta_0 + \sum_{k=1}^{G} \mathbf{x}_{i,k}^{\mathrm{T}} \boldsymbol{\beta}_k, \tag{4}$$

where β_0 is the intercept and $\boldsymbol{\beta}_k \in \mathbb{R}^{p_k}$ is the parameter vector corresponding to the k-th group. For brevity, we denote $\boldsymbol{\beta} \in \mathbb{R}^{p+1}$ as the whole parameter vector, i.e. $\boldsymbol{\beta} = (\beta_0, \boldsymbol{\beta}_1^{\mathrm{T}}, \ldots, \boldsymbol{\beta}_G^{\mathrm{T}})^{\mathrm{T}}$. Since the negative log-likelihood function of the model is a convex function, the estimator $\hat{\boldsymbol{\beta}}_\lambda$ of the model (1) with penalty term (2) is given by the minimizer of the convex function:

$$\begin{aligned} S_\lambda(\boldsymbol{\beta}) = -\textstyle\sum_{i=1}^{n} \log(1 + \exp(-y_i\eta_{\boldsymbol{\beta}}(\mathbf{x}_i))) \\ +\lambda \textstyle\sum_{k=1}^{G} (\sqrt{p_k} \cdot \|\boldsymbol{\beta}_k\|_2), \end{aligned} \tag{5}$$

where tuning parameter $\lambda \geq 0$ controls the amount of penalization.

Because the group lasso selects variables only in group level and does not encourage sparsity within groups, some noise within the selected groups might be retained after the learning stage. Therefore, following [14], we also add a lasso term into the penalty function to yield a solution that achieves the within- and between- group sparsity simultaneously, i.e., many feature groups are exactly zero (thus not selected) and some features (within the non-zero feature group) are also exactly zero. We rewrite (5) as:

$$\begin{aligned} S_\lambda(\boldsymbol{\beta}) = -\textstyle\sum_{i=1}^{n} \log(1 + \exp(-y_i\eta_{\boldsymbol{\beta}}(\mathbf{x}_i))) \\ +\lambda \textstyle\sum_{k=1}^{G} (\sqrt{p_k} \cdot \|\boldsymbol{\beta}_k\|_2) + \zeta\|\boldsymbol{\beta}\|_1, \end{aligned} \tag{6}$$

where the role of ζ is to control the sparse degree within groups.

The simultaneous within- and between- group sparsity makes the model (6) ideal for applications where we are interested in identifying important groups as well as important features within the selected groups.

IV. Model Implementation

In this section, a similarity metric and a feature clustering algorithm are presented, with which the features can be partitioned into groups according to the correlation between features. Then the solution of the regularized regression model with grouped structure is introduced. Finally, the classification implementation of text is given.

A. Algorithm for Generating Grouped Structure

Suppose there is a training matrix $\mathbf{X} \in \mathbb{R}^{p \times n}$ where each column denotes a document with p-dimensional word feature, we can regard each row of $\mathbf{X}$ as the transposition of a n-dimensional word vector (i.e. regard each word as a sample with n dimensionality), the similarity between two word vectors $\mathbf{w}_i$ and $\mathbf{w}_j$ (the transposition of the i-th and j-th row of $\mathbf{X}$) is defined by:

$$s(i,j) = \frac{(\mathbf{w}_i - \bar{\mathbf{w}}_i)^{\mathrm{T}}(\mathbf{w}_j - \bar{\mathbf{w}}_j)}{||(\mathbf{w}_i - \bar{\mathbf{w}}_i)||_2 \times ||(\mathbf{w}_j - \bar{\mathbf{w}}_j)||_2}, \quad (7)$$

where $\bar{\mathbf{w}}_i$ and $\bar{\mathbf{w}}_j$ is the mean value vector of $\mathbf{w}_i$ and $\mathbf{w}_j$, respectively. Therefore, we can utilize a clustering algorithm to divide all words into non-overlapping groups according to the similarity between words.

There exists a large number of clustering algorithms, such as k-means clustering [17], spectral clustering [18], and affinity propagation (AP) [19]. AP is a new clustering method that can automatically determine the number of clusters according to its parameter "preference", and recent studies show that AP offers obvious advantages over existing methods for large scale datasets [19], [20]. Therefore, we employed AP algorithm to partition words into groups.

The AP approach computes two kinds of messages exchanged between data points. The first one is called "responsibility" $r(i,j)$ which is sent from data point i to candidate exemplar point j and reflects the accumulated evidence for how well-suited point j is to serve as the exemplar for point i. The second message is called "availability" $a(i,j)$ which is sent from candidate exemplar point j to point i and it reflects the accumulated evidence for how appropriate it would be for point i to choose point j as its exemplar. At the beginning, all availabilities are initialized to zero. The update equations for $r(i,j)$ and $a(i,j)$ are written as:

$$r(i,j) \leftarrow \; s(i,j) - \max_{j' s.t. j' \neq j} a(i,j') + s(i,j'), \quad (8)$$

$$a(i,j) \leftarrow \begin{cases} \min\{0, r(j,j) + \\ \quad \sum\limits_{i' s.t. i' \notin \{i,j\}} \max\{0, r(i',j)\}\} & i \neq j \\ \\ \sum\limits_{i' s.t. i' \neq j} \max\{0, r(i',j)\} & i = j \end{cases} . \quad (9)$$

In addition, to avoid the oscillations caused by "overshooting" the solution, the responsibility and availability messages are dumped as follows:

$$M_{new} \leftarrow \lambda \times M_{old} + (1-\lambda) \times M_{new}, \quad (10)$$

where M_{new} and M_{old} are the message values from the previous and current iterations, respectively, and $\lambda \in [0,1]$ is the damping factor [19].

The updated procedure may be terminated after a fixed number of iterations, or after changes in the messages fall below a threshold. Then, the exemplar of each point i can be determined by:

$$c_i \leftarrow \arg\max_j \{a(i,j) + r(i,j)\}, \quad (11)$$

and thus, the grouped structure can be constructed by assigning each word into a group.

B. Model Solution

After the grouped structure was constructed, the corresponding regression model can be obtained by solving the minimizer problem of the convex function:

$$S_\lambda(\boldsymbol{\beta}) = \tilde{l}(\boldsymbol{\beta}) + \tilde{\phi}(\boldsymbol{\beta}), \quad (12)$$

where $\tilde{l}(\boldsymbol{\beta})$ and $\tilde{\phi}(\boldsymbol{\beta})$ denote the the empirical loss function and the penalty function of (6) (or (5)), respectively.

Since the term $\tilde{\phi}(\boldsymbol{\beta})$ is non-smooth and more complex than lasso, it is difficult to solve the optimal problem . In this study, we adapt the method described in [21] where an efficient algorithm is designed to solve the grouped structure regularized optimization problem, which has a time complexity comparable to lasso.

One of the key steps to solve (12) is to solve the Moreau-Yosida regularization problem which is defined by:

$$\phi_\lambda(v) = \min_{\boldsymbol{\beta}} \{f(\boldsymbol{\beta}) = \frac{1}{2}||\boldsymbol{\beta} - v||_2^2 + \tilde{\phi}(\boldsymbol{\beta})\}. \quad (13)$$

The Moreau-Yosida regularization has a useful properties, i.e. $\phi_\lambda(v)$ is continuously differentiable despite the fact that $\tilde{\phi}(\boldsymbol{\beta})$ is non-smooth [22], so it can be solved efficiently, and then we can obtain the projection of β in the feasible set. We denote the minimizer of (13) as $\pi_\lambda(\cdot)$, and it can be solved with an efficient method (Algorithm 1 in [21]) which can run fast.

Because the objective function of (12) is a convexity function, if it has an optimal solution $\boldsymbol{\beta}^*$, then there exists:

$$0 \in \tilde{l}'(\boldsymbol{\beta}^*) + \partial\tilde{\phi}(\boldsymbol{\beta}^*), \quad (14)$$

which leads to:

$$0 \in \boldsymbol{\beta}^* - (\boldsymbol{\beta}^* - \tau\tilde{l}'(\boldsymbol{\beta}^*)) + \tau\partial\tilde{\phi}(\boldsymbol{\beta}^*), \; \forall\tau > 0. \quad (15)$$

Hence, $\boldsymbol{\beta}^* = \arg\min_{\boldsymbol{\beta}} \frac{1}{2}||\boldsymbol{\beta} - (\boldsymbol{\beta}^* - \tau\tilde{l}'(\boldsymbol{\beta}^*))||^2 + \tau\tilde{\phi}(\boldsymbol{\beta})$. Since $\pi_\lambda(\cdot)$ is the minimization of (13), we have:

$$\boldsymbol{\beta}^* = \pi_{\lambda_\tau}(\boldsymbol{\beta}^* - \tau\tilde{l}'(\boldsymbol{\beta}^*)), \forall\tau > 0, \quad (16)$$

and thus we can solve the optimal problem of (12) by the fixed point continuation method [23] where the authors show that the method is far superior to that of directly applying the fixed-point iterations.

C. Text Categorization

Given a document vector $\mathbf{d} = (t_1, t_2, \ldots, t_p)^{\mathrm{T}}$, where p is the dimensionality in the term space (here term means distinct word in training set). The $tfidf$ value [24] for each term is defined as:

$$tfidf(t_i, d) = tf(t_i, d) \times idf(t_i), \quad (17)$$

where $tf\,(t_i, d)$ denotes the number of times that t_i occurred in d, and $idf\,(t_i)$ is the inverse document frequency which is defined as $idf(t_i) = \log(n/df(t_i))$, where n is the number of documents in training set and $df\,(t_i)$ denotes the number of documents in training set in which t_i occurs at least once. Then a document can be represented as a vector:

$$\mathbf{x} = (w_1, w_2, \ldots, w_p)^{\mathrm{T}}, \quad (18)$$

where $w_i = tfidf(t_i, d)/\sqrt{\sum_j^n tfidf(t_j, d)^2}$, thus all document vectors in training set can be combined as a term by document matrix $\mathbf{X} \in \mathbb{R}^{p \times n}$.

Since text categorization is a multi-class categorization problem, we use the one-vs-the-rest strategy based on the binary logistic regression models. For each category c, we compute the conditional probability value $\mathbb{P}(y_i = +1|\boldsymbol{\beta}_c, \mathbf{x}_i)$ (3), where $\mathbf{x}_i$ is the vector of the i-th document for classification, y_i is the corresponding label variable, and $\boldsymbol{\beta}_c$ is the learned parameter vector for the category c.

According to the uni-label text corpus (i.e., a document can only be assigned to a unique category), we define the category decision function:

$$Category(\mathbf{x}_i) = \arg\max_c \mathbb{P}(y_i = +1|\boldsymbol{\beta}_c, \mathbf{x}_i). \quad (19)$$

According to the multi-label text corpus (i.e., a document can be assigned to several categories simultaneously), we define the category decision function:

$$Category(\mathbf{x}_i) = \arg_c(\mathbb{P}(y_i = +1|\boldsymbol{\beta}_c, \mathbf{x}_i) > 0.5). \quad (20)$$

V. Experiments

We compared our method with two closely related studies: the ridge logistic regression [7] (denoted by Ridge for convenience), the lasso logistic regression [7] (denoted by Lasso). We denote our method modeled by (12) as Sparse Group Lasso. As a contrast, we also report the performance modeled by (5) and denote it as Group Lasso, which is a special case of our method when we set $\zeta = 0$, i.e., we only select feature in between- group sparsity method.

A. Data Sets

Three popular text categorization benchmarks are tested in our experiments: Reuters-21578, WebKB and 20-newsgroups.

The Reuters-21578 data set[1] is a standard multi-label text categorization benchmark and contains 135 categories. In our experiments, we used a subset of the data collection that includes the 10 most frequent categories among the 135 topics and we call it Reuters-top10. We divided it into the training and test set with the standard "ModApte" version partition. The pre-processed including removing the stop words, switching upper to lower case, and stemming[2].

The WebKB data set contains web pages gathered from university computer science departments. We used the subset called WebKB4[3] including four most populous entity-representing categories. The 20-Newsgroups data set[4] contains approximately 20,000 articles evenly divided among 20 usenet newsgroups.

We also removed the low frequency words (less than three) in these collections. The characters of these collections are summarized in Table I.

TABLE I.
THE DESCRIPTIONS OF DATA SETS (#· DENOTES THE NUMBER OF ·)

Dataset	#training	#test	#feature	#category
Reuters-top10	5,920	2,315	5,585	10
WebKB4	2,777	1,376	7,287	4
20-Newsgroups	11,269	7,505	31,116	20

B. Evaluation Measures

In text categorization, the most commonly used performance measures are recall, precision, and their harmonic mean F_1 [1]. Given a specific category c_i from the category space $\{c_1, \ldots, c_m\}$, the corresponding recall (Re_i), precision (Pr_i) and F_{1i} are defined as follows:

$$Re_i = \frac{TP_i}{TP_i + FN_i}, \quad (21)$$

$$Pr_i = \frac{TP_i}{TP_i + FP_i}, \quad (22)$$

$$F_{1i} = \frac{2 \times Re_i \times Pr_i}{Re_i + Pr_i}, \quad (23)$$

where TP_i (true positives) is the number of documents assigned correctly to class i, FP_i (false positives) is the number of documents that do not belong to class i, but are assigned to this class incorrectly, and FN_i (false negatives) is the number of documents that actually belong to class i, but are not assigned to this class.

The average performance of a binary classifier over multiple categories is derived from the micro-averaged and the macro-averaged. For micro-averaged, the measures are computed globally without categorical discrimination. The micro-averaged recall $\widehat{Re}^U$ and micro-averaged precision $\widehat{Pr}^U$ are defined by:

$$\widehat{Re}^U = \frac{\sum_{i=1}^m |TP_i|}{\sum_{i=1}^m (|TP_i| + |FN_i|)}, \quad (24)$$

$$\widehat{Pr}^U = \frac{\sum_{i=1}^m |TP_i|}{\sum_{i=1}^m (|TP_i| + |FP_i|)}, \quad (25)$$

and the micro-averaged F_1 is defined by:

$$\text{Micro}-F_1 = \frac{2 \times \widehat{Pr}^U \times \widehat{Re}^U}{\widehat{Pr}^U + \widehat{Re}^U}. \quad (26)$$

For macro-averaged, F_1-measure is computed locally over each category c_i first and then the average over all categories is taken:

$$\text{Macro}-F_1 = (\sum_i^m F_{1i})/m. \quad (27)$$

To evaluate the performance overall, we adapt the micro-averaged F_1 and macro-averaged F_1 as the performance measures.

We also report the degree of sparsity for each model, which is defined by:

$$\text{sparsity} = (1 - \frac{\#\text{selected}}{\#\text{whole}}) \times 100\%, \quad (28)$$

where #selected denotes the average number of selected features in the model, and #whole denotes the number of features in the dataset.

[1] http://www.daviddlewis.com/resources/
[2] http://tartarus.org/~martin/PorterStemmer/
[3] http://web.ist.utl.pt/~acardoso/
[4] http://people.csail.mit.edu/jrennie/

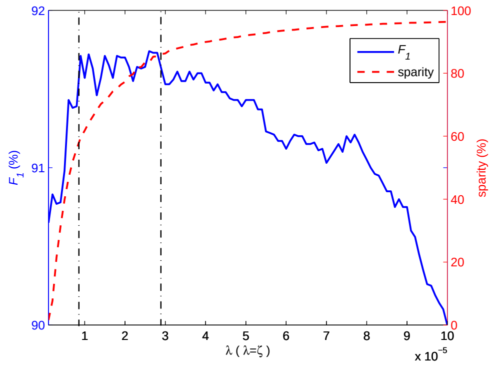

Figure 1. An example of searching parameter (The candidate λ lies in the interval bounded by two dash-dot line. In this interval, F_1 varies slightly; however, the sparsity changes from 50% to 80%, so λ was selected by considering both F_1 and sparsity.)

C. Implementation Details

To compare with the Ridge and Lasso, we used the open-source implementation BBR[5]. We set the parameter "-C" (cross-validation: number of folds) by 10 and the parameter "–autosearch" for enabling self-tuning of the regularization parameter.

The parameter "preference" of AP was assigned by the median of all nonzero similarity values. We used a default damping factor of $\lambda = 0.5$. Consequently, we obtained 783 groups on Reuters-top10, 825 groups on WebKB4, and 3106 groups on 20-Newsgroups.

For each category, we calculated F_1 and sparsity by 10 fold cross-validation on the training set. According to these values, we sought the regularization parameter λ (we simply set $\lambda = \zeta$ in (12) for the computational efficiency). Figure 1 illustrates a searching example for the first category on WebKB4. It is shown that λ might be selected in the interval bounded by two dash-dot line. In this interval, F_1 varies slightly; however, the sparsity changes from 50 to 80%. Therefore, we actually selected λ that achieves the maximum value of $(F_1 + 0.05 * sparsity)$ to favor sparse degree.

D. Results

The experimental results on Reuters-top10 are reported in Table II. The performance of our Group Lasso and our Sparse Group Lasso is slightly better than the Lasso whereas the Lasso reaches the most sparsity among these methods. It shows the Lasso can yield a considerable sparse model and maintain a stable performance where the number of features approaches the number of training samples. In contrast with the Group Lasso, the Sparse Group Lasso obtains a slightly better performance and achieves a more spare solution.

TABLE II.
RESULTS ON REUTERS-TOP10

Methods	$\text{Micro}-F_1$	$\text{Macro}-F_1$	Sparsity
Ridge	94.16	87.59	0.00
Lasso	94.00	88.63	**96.81**
Group Lasso	94.44	**89.51**	81.07
Sparse Group Lasso	**94.52**	**89.51**	86.12

TABLE III.
RESULTS ON WEBKB4

Methods	$\text{Micro}-F_1$	$\text{Macro}-F_1$	Sparsity
Ridge	89.17	87.66	0.00
Lasso	89.32	88.09	**94.85**
Group Lasso	**91.50**	**90.93**	66.17
Sparse Group Lasso	91.06	90.28	85.76

In Table III, the classification results on WebKB4 are presented. The performance of the Lasso is inferior than the Group Lasso and the Sparse Group Lasso. The Group Lasso obtains the best performance, but its sparsity is not satisfied, whereas the Sparse Group Lasso achieves a good tradeoff between the performance and sparsity. Figure 2 illustrates some selected results for category "student" on WebKB4, the middle is some groups in which most words have discriminative meaning, the left groups contain the selected words by the Sparse Group Lasso and the right groups indicate words selected by the Lasso. We can observe that the Sparse Group Lasso obtained more discriminative words than the Lasso.

Table IV shows the results on 20-Newsgroups. The Lasso obtains a poor performance due to the excessive sparsity (only 1.5% words are selected, which means many useful discriminative features may be discarded). The Ridge archives the best performance in term of $\text{Macro}-F_1$; however, it yields a fully dense model. The performance of the Group Lasso is superior to the Lasso but it sparsity is not ideal. Among these methods, the Sparse Group Lasso obtains a relatively satisfied tradeoff between the performance and sparsity.

To sum up, our grouped structure-based methods (i.e., the Group Lasso and the Sparse Group Lasso) achieve a good performance for text categorization. Moreover, the simultaneous within- and between- group sparsity method (i.e. the Sparse Group Lasso) can obtain a good tradeoff between the performance and sparsity in most scenarios.

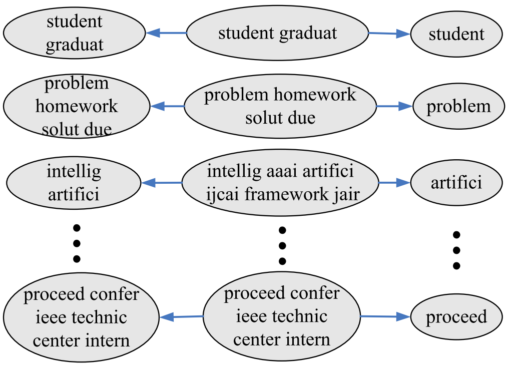
student graduat
student graduat
student
problem homework solut due
problem homework solut due
problem
intellig artifici
intellig aaai artifici ijcai framework jair
artifici
proceed confer ieee technic center intern
proceed confer ieee technic center intern
proceed

Figure 2. An illustration about the selected words for category "student" on WebKB. The middle is some groups in which most words have discriminative meaning, the Sparse Group Lasso obtained more discriminative words (left groups) than the Lasso (right groups). (Note that all words have been stemmed in this figure.)

[5] http://www.bayesianregression.org/

TABLE IV.
RESULTS ON 20-NEWSGROUPS

Methods	Micro$-F_1$	Macro$-F_1$	Sparsity
Ridge	80.80	**80.25**	0.00
Lasso	76.75	76.31	**98.52**
Group Lasso	80.21	79.51	46.70
Sparse Group Lasso	**80.93**	80.17	65.89

VI. CONCLUSIONS

In this paper, we applied a grouped structure-based regularized regression model for text categorization. By developing a feature clustering algorithm, the grouped structure is constructed effectively, which can be used in a logistic regression model to select features in a between- and within- group manner. Then the important groups and important features within the selected groups can be retained during the model fitting. After that, classification is implemented in the logistic regression model. The experimental results show that the proposed method achieves a good tradeoff between the performance and sparsity in most scenarios.

The key of the proposed approach is the combination between grouped structure regularization and the feature clustering, which might set up a bridge between the heuristic knowledge and the statistic regression model.

For further study, we are trying to incorporate more heuristic knowledge of language into the grouped structure to improve the performance.

REFERENCES

[1] F. Sebastiani, "Machine learning in automated text categorization," *ACM Comput. Surv.*, vol. 34, no. 1, pp. 1–47, 2002.

[2] Y. Yang and J. Pedersen, "A comparative study on feature selection in text categorization," in *International Conference on Machine Learning*, 1997, pp. 412–420.

[3] Y. Xu, "A data-drive feature selection method in text categorization," *Journal of Software*, vol. 6, no. 4, pp. 620 – 627, 2011.

[4] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *J. Mach. Learn. Res.*, vol. 3, pp. 1157–1182, 2003.

[5] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. R. Stat. Soc. B*, vol. 58, no. 1, pp. 267–288, 1996.

[6] V. Roth, "The generalized lasso," *IEEE T. Neural. Networ.*, vol. 15, no. 1, pp. 16–28, 2004.

[7] A. Genkin, D. D. Lewis, and D. Madigan, "Large-scale bayesian logistic regression for text categorization," *Technometrics*, vol. 49, pp. 291–304, 2007.

[8] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *J. R. Stat. Soc. B*, vol. 67, no. 2, pp. 301–320, 2005.

[9] S. Aseervatham, A. Antoniadis, E. Gaussier, M. Burlet, and Y. Denneulin, "A sparse version of the ridge logistic regression for large-scale text categorization," *Pattern Recogn. Lett.*, vol. 32, pp. 101–106, 2011.

[10] T. Zhang and F. Oles, "Text categorization based on regularized linear classification methods," *Inform. Retrieval*, vol. 4, pp. 5–31, 2001.

[11] J. Zhang, R. Jin, Y. Yang, and A. Hauptmann, "Modified logistic regression: An approximation to svm and its applications in large-scale text categorization," in *International Conference on Machine Learning*, vol. 20, no. 2, 2003, pp. 888–895.

[12] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," *J. R. Stat. Soc. B*, vol. 68, no. 1, pp. 49–67, 2006.

[13] L. Meier, S. Van De Geer, and P. Bühlmann, "The group lasso for logistic regression," *J. R. Stat. Soc. B*, vol. 70, no. Part 1, pp. 53–71, 2008.

[14] J. Friedman, T. Hastie, and R. Tibshirani, "A note on the group lasso and a sparse group lasso," Department of Statistics,Stanford University, Tech. Rep., 2010.

[15] H. Yang, Z. Xu, I. King, and M. Lyu, "Online learning for group lasso," in *International Conference on Machine Learning*, 2010.

[16] Y. Qian, J. Zhou, M. Ye, and Q. Wang, "Structured sparse model based feature selection and classification for hyperspectral imagery," in *International Geoscience and Remote Sensing Symposium*, 2011, pp. 1771–1774.

[17] J. MacQueen *et al.*, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, vol. 1, no. 14, 1967, pp. 281–297.

[18] A. Pothen, H. Simon, K. Liou, *et al.*, "Partitioning sparse matrices with eigenvectors of graphs." *siam j. matrix anal. applic.*, vol. 11, no. 3, pp. 430–452, 1990.

[19] B. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, no. 5814, pp. 972–976, 2007.

[20] M. Brusco and H. Kohn, "Comment on "clustering by passing messages between data points"," *Science*, vol. 319, no. 5864, p. 726c, 2008.

[21] J. Liu and J. Ye, "Moreau-yosida regularization for grouped tree structure learning," in *Advances in Neural Information Processing Systems*, 2010, pp. 1459–1467.

[22] C. Lemarechal and C. Sagastizabal, "Practical aspects of the moreau-yosida regularization: Theoretical preliminaries," *SIAM J. Optimiz.*, vol. 7, no. 2, pp. 367–385, 1997.

[23] E. Hale, W. Yin, and Y. Zhang, "Fixed-point continuation for l_1-minimization: Methodology and convergence," *SIAM J. Optimiz.*, vol. 19, no. 3, pp. 1107–1130, 2008.

[24] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Inform. Process. Manag.*, vol. 24, no. 5, pp. 513–523, 1988.

Wenbin Zheng is currently a Ph.D. candidate at College of Computer Science and Technology in Zhejiang University, China. He is also a lecturer at China Jiliang University, China. His research interests include machine learning, text and image processing, and pattern recognition.

Yuntao Qian received the Ph.D. degree in signal processing from Xidian University, Xian, in 1996. He was a Postdoctoral Fellow with Northwestern Polytechnical University, Xian, in 1996 to 1998. Since 1998, he has been with Zhejiang University, Hangzhou, China, where he is currently a Professor in computer science. He had been a Visiting Professor with Concordia University, Hong Kong Baptist University, and Carnegie Mellon University, during 1999-2001 and 2006, respectively. His research interests include machine learning, signal and image processing, and pattern recognition.

Minchao Ye received his BS degree in computer science and technology from Sichuan University, China in 2010. He is currently a Ph.D. candidate at College of Computer Science and Technology in Zhejiang University, China. His research interests include machine learning and pattern recognition.