Sparse Poisson Coding for High Dimensional Document Clustering

Chenxia Wu*, Haiqin Yang†‡,Jianke Zhu*, Jiemi Zhang*, Irwin King†‡, and Michael R. Lyu†‡
*College of Computer Science, Zhejiang University, Hangzhou, China
Corresponding author email:jkzhu@zju.edu.cn
† Shenzhen Research Institute, The Chinese University of Hong Kong, Shenzhen, China
‡Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong
Corresponding author email: hqyang@cse.cuhk.edu.hk

***Abstract*—Document clustering plays an important role in large scale textual data analysis, which generally faces with great challenge of the high dimensional textual data. One remedy is to learn the high-level sparse representation by the sparse coding techniques. In contrast to traditional Gaussian noise-based sparse coding methods, in this paper, we employ a Poisson distribution model to represent the word-count frequency feature of a text for sparse coding. Moreover, a novel sparse-constrained Poisson regression algorithm is proposed to solve the induced optimization problem. Different from previous Poisson regression with the family of ℓ_1-regularization to enhance the sparse solution, we introduce a sparsity ratio measure which make use of both ℓ_1-norm and ℓ_2-norm on the learned weight. An important advantage of the sparsity ratio is that it bounded in the range of 0 and 1. This makes it easy to set for practical applications. To further make the algorithm trackable for the high dimensional textual data, a projected gradient descent algorithm is proposed to solve the regression problem. Extensive experiments have been conducted to show that our proposed approach can achieve effective representation for document clustering compared with state-of-the-art regression methods.**

***Index Terms*—document clustering, sparse coding, Poisson regression**

I. Introduction

During past decade, an explosive growth of text contents on Internet makes Web documents become a kind of typical "Big Data" and brings both opportunities and challenges for knowledge discovery, text mining and information retrieval. Among these techniques, document clustering plays very important role in automatic document organization, topic extraction and fast information retrieval or filtering [2], [3], [26]. Typically, text data is represented as a high dimensional binary or count "bag-of-words" vector that brings a great challenge to document clustering. Sparse coding is able to provide a solution by using the unlabeled data to learn a high-level sparse representation of the raw inputs for document clustering [18], [20]. Lots of previous studies have addressed the efficacy of such method in image classification [22], [25].

For a typical sparse coding problem, the feature values are often assumed to be real, which can be described by a Gaussian noise model. Moreover, Gaussian model is mainly designed for continuous data that could take fractional, or negative values [18]. Such assumption is apparently inappropriate for the word-counts data [5], [6], especially, the textual data.

To address this problem, in this paper, we employ a Poisson distribution model on the sparse coding for the frequency data. More specifically, we consider the problem of learning the low frequency count data in the high-dimensional setting. The main challenge of sparse Poisson coding for text clustering is how to effectively model the nonnegative data while selecting the salient features for the succinct model interpretation. Although this problem has been explored in the literature, there still exist some limitations. For web applications [4], Poisson regression is parallelized and implemented under the Hadoop MapReduce framework to provide a scalable and efficient solution for behavioral targeting. The feature selection scheme is quite heuristic, where the important features are selected based on the frequency counted in cookie and the most frequent entities are selected by a predefined threshold. This method may ignore some combination features that occur rarely. In neuroscience [14], ℓ_1-regularized Poisson regression is proposed to learn a sparse representation on the neural activity data. In medical imaging applications [10], [11], sparsity-based penalties following the idea of compressed sensing [8] are proposed to seek a sparse Poisson regression model to reconstruct the true function of the photon intensity. These methods have to specify a regularization parameter to control the sparse level of the solution in selecting the important features. However, the range of the parameter is relatively large and the relationship between the sparsity level of the solution and the parameter is not directly evident. The insufficiency of previously proposed work motivates our further exploration on the sparse Poisson regression models in this work.

To overcome the above issues, we propose a novel Poisson regression model, namely sparsity-constrained Poisson regression (SCPR), to build a linear model for the frequency data while providing the sparsity solution for the salient feature selection.

We highlight the contributions of our work in the following:

- Firstly, we employ a Poisson distribution model to well describe the word-count feature of a text in sparse coding, which can provide a high-level feature for document clustering.
- Secondly, we induce a sparse prior to Poisson coding by adopting a sparsity ratio, which is different from previously proposed sparsity constraints [23], [24]. The

978-1-4799-1293-3/13/$31.00 ©2013 IEEE

sparsity ratio is borrowed from the sparseness constraints for non-negative matrix factorization (NNMF) [12], [13] and utilizes both the ℓ_1-norm and ℓ_2-norm on the weight. More importantly, the ratio is bounded in the range of 0 and 1, which clearly indicates the sparsity level of the solution. It is more intuitive and easier to be set in real-world applications. Also, we investigate how to utilize the sparsity ratio and elaborate different settings for it. We therefore propose the SCPR with an equal constraint to maintain the learned weight at a desired sparsity level.

- Finally, in order to make SCPR applicable for high dimensional text data input, we design a projected gradient descent algorithm for SCPR. The algorithm is very efficient and scales with the average number of non-zero elements, which is much less burden than the original Poisson regression model counting all features.

II. Poisson Regression for Sparse Coding

In this section, we will introduce how to employ Poisson regression to solve a sparse coding problem [7], [9], [17], [22], [25].

We first review the typical sparse coding problem. Let $\mathbf{x} \in \mathbb{R}^n$ be the raw input feature vector. Sparse coding aims to find a set of basis vectors $\mathbf{B} = \{\mathbf{b}_j \in \mathbb{R}^n\}_{j=1}^d$ and the sparse representations/coefficients $\mathbf{z} \in \mathbb{R}^d$ with respect to the basis for $\mathbf{x}$. Moreover, it is typically based on a Gaussian noise model: $\mathbf{P}(\mathbf{x}|\mathbf{B},\mathbf{z}) = \mathbf{N}(\sum_j^d z^{(j)}\mathbf{b}_j, \sigma^2 I)$, where each feature is assumed independent and identically distributed, $z^{(j)}$ is the j-th element of $\mathbf{z}$ and σ^2 is fixed. A sparse prior $\mathbf{P}(\mathbf{z}) \propto \prod_j \exp(-\lambda|z^{(j)}|)$ is assumed to penalize the nonzero representations. Given an unlabeled sample $\mathbf{x}$, the basis vectors and the sparse representations are obtained by the MAP optimization problem:

$$\min_{\mathbf{B},\mathbf{z}} \quad \frac{1}{2\sigma^2}\|\mathbf{x} - \sum_j^d z^{(j=1)}\mathbf{b}_j\|_2^2 + \lambda\sum_{j=1}^d |z^{(j)}|. \tag{1}$$

The above problem can be efficiently solved by the alternative minimization over $\mathbf{B}$ and $\mathbf{z}$ variables [17]. Also, the basis vectors can be efficiently selected as the cluster centers by running k-means clustering on all the data samples [15] or simply selected by randomly sampling from the data [21]. Given a fixed $\mathbf{B}$, $\mathbf{z}$ can be obtained by solving a regression problem by minimizing the objective for both training and the new input data samples.

The probabilistic model for the above problem assumes that the input data features are real-valued, which is typically described by a Gaussian noise model. Obviously, it is inappropriate for the textual data with word-counts frequency $\mathbf{x} \in \{0, 1, 2, ...\}^d$, which may be poorly modeled by a continuous Gaussian distribution [18]. We try to address this problem by introducing the Poisson distribution model in sparse coding on the frequency data. Given a text document $\mathbf{x}$, we use the following Poisson distribution model:

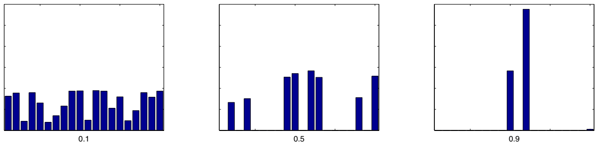


Fig. 1. Illustration of different sparsity levels (0.1, 0.5, and 0.9). At low level of sparsity (left), the weight is dense. At high level (right), most of elements are zeros and only a few take large values.

$$\begin{aligned}\mathbf{P}(x^{(i)}|\mathbf{B},\mathbf{z}) &= \text{Poisson}(x^{(i)}|\sum_j^d z^{(j)} b_j^{(i)}) \\ &= \frac{\mu^{(i)x^{(i)}}\exp(-\mu^{(i)})}{x^{(i)}!},\end{aligned} \tag{2}$$

where $b_j^{(i)}$ is the i-th element of $\mathbf{b}_j$ and $\sum_j^d z^{(j)} b_j^{(i)}$ is simply denoted as $\mu^{(i)}$. As each feature is assumed to be independent and identically distributed, $\mathbf{P}(\mathbf{x}|\mathbf{B},\mathbf{z}) = \prod_i^n \text{Poisson}(x^{(i)}|\mu^{(i)})$. Note that we assume both the basis vector and the new representation vector $\mathbf{z}$ to be non-negative, which imposes a positive effect onto one-unit change in the word count for the expected target word count [4]. Given an unlabeled sample $\mathbf{x}$ and the fixed basis vectors $\mathbf{B}$, the sparse representations $\mathbf{z}$ can be obtained by the MLE optimization problem:

$$\max_{\mathbf{z}} \quad \mathcal{L} = \sum_{i=1}^n (x^{(i)}\log(\mu^{(i)}) - u^{(i)} - \log(x^{(i)}!)). \tag{3}$$

The above log-likelihood is a concave function on $\mathbf{z}$ and therefore guarantees the global optimal solution for Poisson regression. Various algorithms, e.g., gradient descent, can be adopted to seek its optimal solution. The derivative of the log-likelihood with respect to $z^{(j)}$ is

$$\frac{\partial\mathcal{L}}{\partial z^{(j)}} = \sum_{i=1}^n \left(\frac{x^{(i)}}{\mu^{(i)}} b_j^{(i)} - b_j^{(i)}\right). \tag{4}$$

The multiplicative rule [16] is used to update the coefficient vector

$$z^{(j)} \leftarrow z^{(j)} \frac{\sum_{i=1}^n \frac{x^{(i)}}{\mu^{(i)}} b_j^{(i)}}{\sum_{i=1}^n b_j^{(i)}}. \tag{5}$$

III. Sparse Poisson Coding

In this section, we consider how to utilize the sparsity prior in sparse coding problem. Then we propose an effective sparsity-constrained Poisson regression (SCPR) approach with a given sparsity level. To solve this problem, we design an efficient algorithm based on the projected gradient descent and sketch its average computational cost.

A. Sparse Poisson Regression

A sparse prior $\mathbf{P}(\mathbf{z})$ is assumed to penalize the nonzero representations. Then, the sparse representations $\mathbf{z}$ are obtained by the MAP optimization problem:

$$\min_{\mathbf{z}} \quad -\mathcal{L} + \lambda \text{Pen}(\mathbf{z}), \tag{6}$$

where $\mathcal{L}$ is the log-likelihood. $\text{Pen}(\mathbf{z})$ is based on the sparse prior, which can be ℓ_1-regularization $\|\mathbf{z}\|_1$ [14] or the hybrid Huber penalty [10], [11].

B. Sparsity Ratio

Numerous measures can be used to evaluate for the sparsity level of the coefficients. A good indicator function can map a vector from $\mathbb{R}_+^d$ to $[0, 1]$ and quantify how much energy maintain on the component of the coefficient. An ideal one is to have only one non-zero element for the sparse vector and with all elements non-zero for the least sparse case.

Hence, we adopt the sparsity ratio defined in [13] for NNMF in this paper:

$$\text{spr}(\mathbf{z}) := \frac{1}{\sqrt{d}-1}\left(\sqrt{d} - \frac{\|\mathbf{z}\|_1}{\|\mathbf{z}\|_2}\right). \tag{7}$$

In Eq. (7), the defined sparsity ratio is different from the previously proposed sparse Poisson regression mainly utilizing the ℓ_1 regularization. Based on the relationship between the ℓ_1 norm and the ℓ_2 norm. Moreover, this ratio contains a good property, bounding in the range of 0 and 1. We summarize this property of the sparsity ratio in the following proposition:

Proposition III.1. $\forall \mathbf{z} \in \mathbb{R}_+^d$, $\mathbf{z} \neq \mathbf{0}$, *we have*

$$0 \leq \text{spr}(\mathbf{z}) \leq 1. \tag{8}$$

The above proposition can be proved by the following relationships:

$$\frac{1}{\sqrt{d}}\|\mathbf{z}\|_1 \leq \|\mathbf{z}\|_2 \leq \|\mathbf{z}\|_1. \tag{9}$$

The first inequality follows the Cauchy-Schwarz inequality. The second inequality is obvious when putting square on both sides. Although Proposition III.1 is valid for negative $\mathbf{z}$, we still keep the non-negative condition. This is mainly due to that we only consider non-negative coefficient vectors. In Eq. (9), the lower bound is reached when a vector with equal non-zero elements. On the other hand, the upper bound is obtained when a vector with all but one vanishing elements. Hence, this ratio is intuitive and easy to set the sparsity level of a given vector $\mathbf{z}$, see more illustrated examples in Fig. 1.

C. Proposed Model

We consider how to employ the above defined sparsity ratio to obtain the sparse solution. An intuitive setting is to bound the sparsity ratio as follows:

$$\text{spr}(\mathbf{z}) \leq r. \tag{10}$$

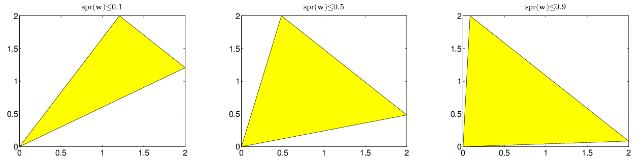
An advantage of this setting is to maintain the convexity of the domain. This can be shown that the domain of the learned coefficient in Eq. (10) is equivalent to

$$\|\mathbf{z}\|_2 \leq \frac{1}{c_{d,r}}\|\mathbf{z}\|_1,$$

where

$$c_{d,r} = \sqrt{d}(1-r) + r. \tag{11}$$

This is exactly a second-order cone, or Lorentz cone [1]. Combining the restriction of $\mathbf{z} \in \mathbb{R}_+^d$, i.e., $\mathbf{z} \geq \mathbf{0}$, we can define the domain of learned coefficient in a convex set. Fig. 2 shows the different examples.



(a) 0.1 (b) 0.5 (c) 0.9

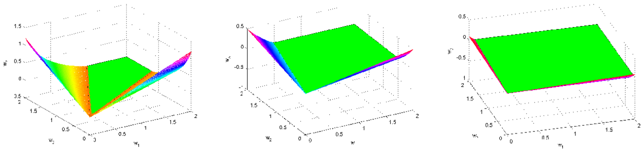
Fig. 2. Illustration of the set of the learned weight bounded on different levels (0.1, 0.5, and 0.9). The larger the bounded sparsity ratio, the larger the domain of learned weight is.

It seems that we can define the learned coefficient by bounding the sparsity ratio as in Eq. (10) and yield a convex optimization problem. As illustrated in Fig. 2, a large bound, i.e., $\text{spr}(\mathbf{z}) = 1$, the defined domain of the learned coefficient will recover the whole domain of the learned coefficient, which is exactly the original domain of Poisson regression. Moreover, defining $\mathbf{z}$ on the conic set in Eq. (10) also cannot yield sparse solution.

Hence, we borrow the idea of the sparsity constraints proposed for NNMF in [13] and propose the sparsity-constrained Poisson regression (SCPR) as follows:

$$\max_{\mathbf{z} \in \mathbb{R}_+^d} \quad \mathcal{L}, \quad \text{s.t.} \quad \text{spr}(\mathbf{z}) = r. \tag{12}$$

Note that our proposed SCPR requires the learned coefficient at a certain sparsity level via the pre-defined parameter r. For the different sparsity levels, we can refer to the charts shown in Fig. 3. It can be found that the larger $\text{spr}(\mathbf{z})$, the smaller valid set is. Extremely, the set consists of several isolated points on the corresponding axis, when the sparsity ratio is set to one.



(a) $\text{spr}(\mathbf{w}) == 0.1$ (b) $\text{spr}(\mathbf{w}) == 0.5$ (c) $\text{spr}(\mathbf{w}) == 0.9$

Fig. 3. Illustration of the sparsity-constraint set in a 3D wireframe mesh for different sparsity levels (0.1, 0.5, and 0.9). The set consists of the mesh except the green region. The set becomes small as the sparsity ratio increases.

D. Algorithms

In order to make our proposed SCPR applicable for the high-dimensional textual data, we design an efficient projected gradient descent algorithm for the minimization problem in SCPR. The whole algorithm consists of two main steps: 1) the first step mainly follows the "multiplicative update rule" as in Eq. (5) to update the coefficient vector; 2) the second step is to

Algorithm 1 Projected gradient descent for SCPR

```
Input: The raw feature $\mathbf{x}$ and the fixed basis vectors $\mathbf{B}$, the sparsity ratio $r$;
Output: The coefficient vector of SCPR $\mathbf{z}$.
Compute $d = \text{dim}(\mathbf{z})$;
Compute $c_{d,r}$ by Eq. (11);
Initialize $\mathbf{z}$ to random positive vector;
Set $L2$ to the square of $\ell_2$-norm on $\mathbf{z}$, i.e., $L2 = \|\mathbf{z}\|_2^2$;
Set $L1$ to the $\ell_1$-norm value corresponding to the desired level of sparsity, i.e., $L1 = c_{d,r} \times \|\mathbf{z}\|_2$;
Call Projection($\mathbf{z}$, $L1$, $L2$) to update $\mathbf{z}$;
repeat
    Calculate $\mathbf{z}$ by Eq. (5);
    Set $L2$ to the square of $\ell_2$-norm on $\mathbf{z}$, i.e., $L2 = \|\mathbf{z}\|_2^2$;
    Set $L1$ to the $\ell_1$-norm value corresponding to the desired level of sparsity, i.e., $L1 = c_{d,r} \times \|\mathbf{z}\|_2$;
    Call Projection($\mathbf{z}$, $L1$, $L2$) to update $\mathbf{z}$;
until converge.
```

Algorithm 2 Projection($\mathbf{z}$, $L1$, $L2$)

```
Objective: Find the closet non-negative vector to a vector $\mathbf{z}$ with a given $\ell_1$ norm, $L1$, and a given square of the $\ell_2$-norm, $L2$.
1:  Compute $d = \text{dim}(\mathbf{z})$;
2:  Compute $S = \{j : z^{(j)} \leq 0\}$;
3:  Compute $v^{(j)} = \begin{cases} z^{(j)} + \frac{L1 - \sum_t z^{(t)}}{d - \text{size}(S)} & \text{if } j \notin S \\ 0 & \text{if } j \in S \end{cases}$
4:  loop
5:      Compute $m^{(j)} = \begin{cases} L1/(d - \text{size}(S)) & \text{if } j \notin S \\ 0 & \text{if } j \in S \end{cases}$
6:      Update $\mathbf{v} = \mathbf{v} + \eta(\mathbf{v} - \mathbf{m})$, where $\eta$ is the non-negative root of the quadratic equation, $\|\mathbf{v} + \eta(\mathbf{v} - \mathbf{m})\|_2^2 = L2$;
7:      if all elements of $\mathbf{v}$ are non-negative then
8:          return $\mathbf{v}$.
9:      end if
10:     Update $S = \{j : v^{(j)} \leq 0\}$;
11:     Update $v^{(j)} = 0, \forall j \in S$;
12:     Compute $s = (\sum_t v^{(t)} - L1)/(d - \text{size}(S))$;
13:     Update $v^{(j)} = v^{(j)} - s, \forall j \notin S$;
14: end loop
```

project the coefficient onto the constraint space to achieve the desired level of sparsity. This procedure is summarized into Algorithm 1.

In Algorithm 1, the function **Projection**($\mathbf{z}$, $L1$, $L2$) has to be called several times, which is to project $\mathbf{z}$ with the corresponding $L1$-norm and the square of $L2$-norm to achieve the desired sparsity. The procedure is defined in Algorithm 2.

For Algorithm 2, we first remove those elements with the value being zero at line 2. At line 3, a point on a hyperplane $\sum_t v^{(t)} = L1$ is initialized. Then, we move from the center of the sphere towards the initialized point to satisfy the ℓ_2 constraint, where the center is defined by the point with all elements being equal in the updated index. If the updated point is non-negative for all the elements, and then the algorithm is terminated. Otherwise, we reset those elements with negative values to zero, and project the point back onto the hyperplane with $\sum_t v^{(t)} = L1$.

Time complexity analysis. Comparing to the original Poisson regression model, the proposed SCPR requires invoking the function **Projection** at each iteration, which incurs some computation efforts. In the worst case, Algorithm 2 may take as many iterations as the number of coefficients dimension, i.e., dim($\mathbf{z}$), to converge to an optimal solution. However, the algorithm converges much faster in practice. It just needs about four iterations for the worse case and one or two iterations for the optimal solution at average.

Hence, the number of iterations in the function **Projection** can be considered as a constant. Moreover, the computation cost for the function **Projection** is proportional to the number of non-zero elements (NNZs) in the learned coefficient. Additionally, the number of outer iterations required by Algorithm 1 is much smaller than the original Poisson regression model due to the sparse solution.

In summary, Poisson regression has to update the coefficient for all elements due to non-sparsity. The number of outer iterations is proportional to the dimension of the raw features and the number of coefficient dimension. We abstract it as $c(n, d)$ and obtain the time cost of PR as $\mathcal{O}(c(n, d) \times d)$. SCPR requires $c(n, d)$ outer iterations, which is nearly a constant, and several times to invoking the function **Projection**. At for each iteration, SCPR only needs to update those non-zero elements. Hence, the average run time cost for SCPR is in the order of $\mathcal{O}(c(n) \times \text{Avg(NNZs)})$, which is much smaller than the original Poisson regression model.

IV. Experiments

To study the efficacy and the merits of our proposed SCPR method in different perspectives, we evaluate the document clustering performance using the different regression approaches in sparse coding for learning the coefficients. All the experiments are conducted on a notebook computer with Inter i3-3110M CPU@2.40GHz and 4GB memory.

A. Sparse Coding for Document Clustering

In this section, we evaluate the performance of sparse coding for document clustering. To show the efficacy of our algorithm, we study the Poisson regression approaches with different sparse prior and regularization.

1) Dataset: We conduct the performance evaluations on the TDT2 document corpora [3], which consists of the data collected during the first half of $1,998$ and taken from six sources, including two newswires (APW and NYT), two radio programs (VOA and PRI) and two television programs (CNN and ABC). It is composed of $11,201$ on-topic documents which are classified into 96 semantic categories. In this experiment, those documents appearing in two or more categories were removed, and only the largest 30 categories were kept, thus leaving us with $9,394$ documents in total[1]. In the dataset, the stop words are removed and each document is represented as a $36,771$-dimensional TF-IDF vector.

2) Evaluation Measure: In our experiments, the basis for sparse coding is firstly selected by randomly sampling from the data [21]. The raw feature of each document is fed into the different regularized regression approaches in order to obtain the sparse representations, which are further employ to cluster the documents. The clustering result is evaluated by comparing the estimated label for each document using k-means clustering algorithm. Two typical metrics are used to measure the performance. The first metric is the accuracy (AC) [2]. Given a document $\mathbf{x}_i$, let $\hat{h}_i$ and h_i be the estimated cluster label and the label provided by the corpus, respectively. The AC is defined as follows:

$$AC = \frac{\sum_{i=1} N\delta(h_i, \mathrm{map}(\hat{h}_i))}{N}, \tag{13}$$

where N is the total number of documents. $\delta(x;y)$ is the delta function, and $\mathrm{map}(\hat{h}_i)$ is the permutation mapping function that maps each cluster label $\hat{h}_i$ to the equivalent label from the data corpus. The best mapping can be found by using the Kuhn-Munkres algorithm [19].

Another metric is the normalized mutual information (NMI) metric [2]. Let C denote as the set of ground truth clusters and $\hat{C}$ as the cluster set obtained from clustering algorithm. The mutual information metric $MI(C,\hat{C})$ is defined as follows:

$$MI(C,\hat{C}) = \sum_{c_i \in C, c'_j \in \hat{C}} p(c_i, c'_j) \cdot \log_2 \frac{p(c_i, c'_j)}{p(c_i) \cdot p(c'_j)}, \tag{14}$$

where $p(c_i), p(c'_j)$ are the probabilities that a document arbitrarily selected from the corpus belongs to the clusters c_i and c'_j, respectively. $p(c_i, c'_j)$ is the joint probability that the arbitrarily selected document belongs to the cluster c_i as well as c'_j at the same time. We further employ the normalized mutual information (NMI):

$$NMI(C,\hat{C}) = \frac{MI(C,\hat{C})}{\max(H(C), H(\hat{C}))}, \tag{15}$$

where $H(C), H(\hat{C})$ are the entropies of C and $\hat{C}$, respectively. From the definition, we know $NMI(C,\hat{C})$ ranges from 0 to 1. If two sets of clusters are identical, NMI is equal to one. If they are independent, NMI is set to zero.

3) Performance Evaluation: We compare our proposed SCPR algorithm with several algorithms: Gaussian regression ℓ_1 norm regularizer (GR-ℓ_1), the original Poisson regression without sparse prior (PR), Poisson regression with ℓ_1 norm regularizer (PR-ℓ_1), Poisson regression with Recursive Dyadic Partitions (RDPs) regularizer (PR-RDP), Poisson regression with translationally-invariant (cycle-spun) RDPs (PR-TI), Poisson regression with Total Variation semi-norm regularizer (PR-TV). To facilitate the fair comparisons, we directly adopt the implementation of these reference methods with the recommended settings [10]. In our experiments, the dictionary size is set to $d = 1000$ and $d = 2000$, respectively. In each case, the sparsity level r is set to 0.5 and 0.7, respectively, according to the cross-validation.

Fig. 4 plots the clustering results. Table I reports the coding time when $d = 1000$. It can be easily found that our presented SCPR approach obtains better clustering performance compared with other methods. This is because our algorithm not only reduces the reconstruction error but also captures the same basis for the similar documents using a fitted sparsity level. Though RDPs and RDP-TI constraints can greatly improve the regression speed, their performances decrease at the same time. Except from these two fast regression methods, other approaches with the sparsity prior show better performance than the raw regression and the Poisson regression-based sparse coding outperforms the Gaussian regression-based sparse coding. This demonstrates that the sparsity prior would capture the salient high-level features of the text to improve the representation ability and Poisson distribution can better describe the word-count feature of the text data. Among these effective sparse Poisson coding methods SCPR method performs best with the least computation time.

V. Conclusion

In this paper, we introduced the Poisson distribution model to represent the word-count textual feature in sparse coding. A novel sparse-constrained Poisson regression algorithm was proposed to solve the induced optimization problem. We have defined a sparsity ratio which employed both ℓ_1-norm and ℓ_2-norm on the learned weight. To further make the algorithm applicable for the high dimensional textual data, we designed a projected gradient descent algorithm to solve the regression problem. We have conducted the regression experiments on the synthetic data to show the improvement of our presented SCPR algorithm compared with the original Poisson regression. The clustering experiment on the high-dimensional textual data indicated that our proposed approach is able to learn the better sparse representation with faster speed for document clustering compared with the state-of-the-art regression methods.

Acknowledgment

The work was supported by the Basic Research Program of Shenzhen (Project No. JCYJ20120619152419087 and JC201104220300A), the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. CUHK 413212 and CUHK 415212), and National Natural Science Foundation of China under Grants (61103105 and 91120302).

References

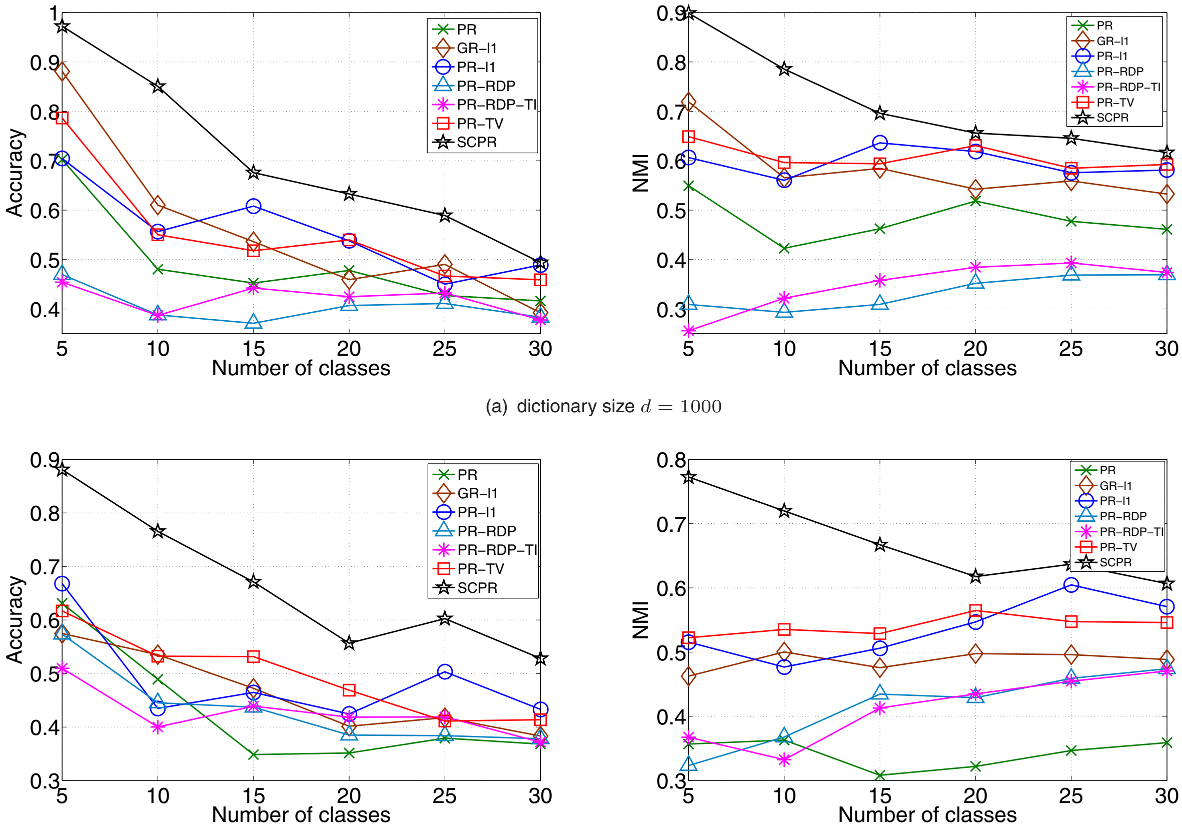
[1] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[2] D. Cai, X. He, and J. Han. Document clustering using locality preserving indexing. *IEEE TKDE*, 17(12):1624–1637, 2005.

[3] D. Cai, X. He, and J. Han. Locally consistent concept factorization for document clustering. *IEEE TKDE*, 23(6):902–913, 2011.

[4] Y. Chen, D. Pavlov, and J. F. Canny. Behavioral targeting: The art of scaling up simple algorithms. *ACM TKDD*, 4(4):17, 2010.

[1] http://www.cad.zju.edu.cn/home/dengcai/Data/TextData.html



(a) dictionary size $d = 1000$

(b) dictionary size $d = 2000$

Fig. 4. Illustration of the clustering results.

TABLE I
AVERAGE TIME PER DOCUMENT OF SPARSE CODING WITH $d = 1000$

approaches	PR	GR-ℓ_1	PR-ℓ_1	PR-RDPs	PR-RDP-TI	PR-TV	SCPR
time/doc.(sec.)	17.28	0.41	0.53	0.10	0.15	0.88	0.49

[5] C. Cheng, H. Yang, , M. R. Lyu, and I. King. Where you like to go next: Successive point-of-interest recommendation. In *IJCAI*, Beijing, China, 2013.

[6] C. Cheng, H. Yang, I. King, and M. R. Lyu. Fused matrix factorization with geographical and social influence in location-based social networks. In *AAAI*, Toronto, Canada, 2012.

[7] P. C. Cosman, R. M. Gray, and M. Vetterli. Vector quantization of image subbands: a survey. *IEEE TIP*, 5(2):202–225, 1996.

[8] D. L. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306, 2006.

[9] J. C. Gemert, J.-M. Geusebroek, C. J. Veenman, and A. W. Smeulders. Kernel codebooks for scene categorization. In *ECCV*, 2008.

[10] Z. Harmany, R. Marcia, and R. Willett. This is spiral-tap: Sparse poisson intensity reconstruction algorithms - theory and practice. *IEEE TIP*, 21(3):1084–1096, 2012.

[11] Z. T. Harmany, R. F. Marcia, and R. Willett. Sparsity-regularized photon-limited imaging. In *ISBI*, pages 772–775, 2010.

[12] M. Heiler and C. Schnörr. Learning sparse representations by non-negative matrix factorization and sequential cone programming. *Journal of Machine Learning Research*, 7:1385–1407, 2006.

[13] P. O. Hoyer. Non-negative matrix factorization with sparseness constraints. *Journal of Machine Learning Research*, 5:1457–1469, 2004.

[14] R. C. Kelly, M. A. Smith, R. E. Kass, and T. S. Lee. Accounting for network effects in neuronal responses using l1 regularized point process models. In *NIPS*, pages 1099–1107, 2010.

[15] S. Kumar, M. Mohri, and A. Talwalkar. On sampling-based approximate spectral decomposition. In *ICML*, 2009.

[16] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. In *NIPS*, pages 556–562, 2000.

[17] H. Lee, A. Battle, R. Raina, and A. Y. Ng. Efficient sparse coding algorithms. In *NIPS*, pages 801–808, 2007.

[18] H. Lee, R. Raina, A. Teichman, and A. Y. Ng. Exponential family sparse coding with applications to self-taught learning. In *IJCAI*, pages 1113–1119, 2009.

[19] L. Lovász and M. Plummer. *Matching Theory*. Akadémiai Kiadó, Budapest, 1986.

[20] R. Raina, A. Battle, H. Lee, B. Packer, and A. Y. Ng. Self-taught learning: transfer learning from unlabeled data. In *ICML*, pages 759–766, 2007.

[21] D. Wang, S. C. Hoi, Y. He, and J. Zhu. Retrieval-based face annotation by weak label regularized local coordinate coding. In *ACM international conference on Multimedia*, pages 353–362, 2011.

[22] J. Wang, J. Yang, K. Yu, F. Lv, T. S. Huang, and Y. Gong. Locality-constrained linear coding for image classification. In *CVPR*, pages 3360–3367, 2010.

[23] H. Yang, M. R. Lyu, and I. King. Efficient online learning for multi-task feature selection. *ACM TKDD*, 7(2):6, 2013.

[24] H. Yang, Z. Xu, J. Ye, I. King, and M. R. Lyu. Efficient sparse generalized multiple kernel learning. *IEEE TNN*, 22(3):433–446, March 2011.

[25] J. Zhang, C. Wu, D. Cai, and J. Zhu. Bilevel visual words coding for image classification. In *IJCAI)*, 2013.

[26] T. Zhang, Y.-Y. Tang, B. Fang, and Y. Xiang. Document clustering in correlation similarity measure space. *IEEE TKDE*, 24(6):1002–1013, 2012.