Abstract—Feature selection plays an important role in machine learning applications. Especially for text data, the high-dimensional and sparse characteristics will affect the performance of feature selection. In this paper, an unsupervised feature selection algorithm through Random Projection and Gram-Schmidt Orthogonalization (RP-GSO) from the word co-occurrence matrix is proposed. The RP-GSO has three advantages: (1) it takes as input dense word co-occurrence matrix, avoiding the sparseness of original document-term matrix; (2) it selects “basis features” by Gram-Schmidt process, guaranteeing the orthogonalization of feature space; and (3) it adopts random projection to speed up GS process. We did extensive experiments on two real-world text corpora, and observed that RP-GSO achieves better performance comparing against supervised and unsupervised methods in text classification and clustering tasks.

I. INTRODUCTION

For text data, high-dimensionality and sparseness are two primary characteristics. The high dimensionality often causes “curse of dimensionality”, increases training time, and affects performance of learning models [11], [14], [12]. Feature selection (FS) plays an important role, and they can be divided into supervised and unsupervised ones based on whether the label information is available, such as forward selection [6], t-test [15], MCFS [4].

Recently, some researchers have designed provably efficient algorithms to statistically recover topic model parameters [2], [12], [7]. For example, Donoho and Stodden [8] proposed the “Separability” concept in non-negative matrix factorization (NMF), which assumes that every topic contains at least one anchor word that has non-zero probability only in that topic. From matrix perspective, the assumption demonstrates that there exists some basis vectors residing in the original space, which can span the whole space.

In the above approaches, “anchor words” are selected from term-document matrix $X$ or word co-occurrence matrix $Q$. We observe that the transformation from $X$ to $Q$ is more favorable for feature selection algorithms, because the transformed space is more dense and suit to find orthogonal basis vectors. As shown in Figure 1, we selected two most frequent words from Reuters corpus, and the row identifications are 2909 and 2931, respectively. Figure 1(a) shows the scatter diagram of two row vectors in term-document space, i.e., $X(2909,:)$ and $X(2931,:)$, in which the green points stands for $X(2931,:)$, and the red points are $X(2909,:)$. It is obvious that two row vectors are partially overlapping in original term-document space. Figure 1(b), however, demonstrates the change after we transform $X$ into $Q$. That is, two vectors are almost wholly overlapping in word co-occurrence space. Therefore, the matrix $Q$ not only avoids the sparseness of matrix $X$, but also conquers too narrow projection distribution during orthogonalization process in our algorithm.

Inspired from the “separability” in NMF, and from the “orthogonalization” in matrix, we propose an unsupervised feature selection algorithm through Random Projection and Gram-Schmidt Orthogonalization (RP-GSO), which detects basis features from normalized empirical word co-occurrence matrix using Gram-Schmidt orthogonalization [6], [9], and employs random projection to speed up the above orthogonalization process. In this paper, we made the following
contributions:

- The proposed RP-GSO algorithm takes as input dense word co-occurrence matrix, rather than sparse document-term matrix. The modification of input not only avoids the sparseness of matrix, but also conquers too narrow projection distribution during GS orthogonalization process.

- Our algorithm applies linear dependence and orthogonalization of vector space to detect basis vectors, which can be regarded as “basis features”.

- Extensive experiments in real-world text classification and clustering tasks demonstrate the superiority of our proposed algorithm, compared against the state-of-art unsupervised and supervised feature selection algorithms.

II. RELATED WORK

Here we briefly review recent unsupervised feature selection approaches, especially from topic model perspective.

Actually, researchers in theoretical computer science have been focused on feature selection algorithms from matrix theories perspective. For example, early in 1989, Chen et al. [6] first applied Gram-Schmidt orthogonalization projection in feature selection area, and proposed the supervised forward selection method. Recently, Donoho and Stodden proposed “Separability” concept in non-negative matrix factorization. Along this line, Arora et al. proposed topic discovery algorithm based on anchor words and word co-occurrence probability matrix with provable guarantees [2]. Kumar et al. directly find extreme rays of the conical hull of a finite set of vectors, which form word-document matrix of the corpus [12]. Ding et al. proposed an algorithm based on cross-document word-frequency patterns, and presented two efficient methods to detect “novel words”, i.e., data-dependent and random projections [7], which achieved similar performance in their experiments.

III. RP-GSO ALGORITHM

Given a corpus of documents $D$, and $D$ are composed of $M$ documents with a vocabulary $V$ size of $W$. For each document $d_i (1 \leq i \leq M)$ in $D$, let $X_i$ be the column vector in $\mathbb{R}^W$, i.e., the classic “bags of words” modeling paradigm, such that the $j$-th ($1 \leq j \leq W$) entry $X_{ij}$ is the term frequency of word $w_j$ in $d_i$. Similar to Arora et al. [2], we use $X_i = X_i / \sqrt{n_d(n_d-1)}$ as the normalized vector of $X_i$, and the diagonal matrix is $\bar{X}_i = \text{Diag}(X_i)/n_d(n_d-1)$. Then we collect all the column vectors $X_i$ to form a large sparse matrix $\bar{X}$, and sum all $X_i$ to get the diagonal matrix $\bar{X}$, then we get word co-occurrence matrix $Q$ by Eq. 1.

$$Q = \frac{1}{M} (\bar{X} \bar{X}^T - \bar{X}),$$  \hspace{1cm} (1)

where $Q$ is a $W \times W$ square matrix. In the infinite data case where we collect infinitely many documents, $Q$ would be the second-moment matrix of $X$. Meanwhile, from probabilistic perspective, each entry $Q_{ij}$ stands for the empirical co-occurrence probability between word $i$ and word $j$ in the corpus, i.e., $Q_{ij} = p(w_i, w_j)$. After removing scalar parameter $\frac{1}{M}$, the matrix $-Q$ is similar to graph Laplacian matrix $L$ in spectral learning if we consider each word as each document, and consider co-occurrence frequency as a distance similarity function between words.

The theoretical basis of our algorithm is separability assumption proposed by Donoho [8], which assumes that there exists some words perfectly indicating the corresponding topic.

Separability assumption. The word-topic matrix $\beta_{w \times K}$ is $p$-separable for $p > 0$ if for each topic $k$, there is some word $i$ such that $\beta_{i,k} \geq p$ and $\beta_{i,k'} = 0$ for $k' \neq k$.

Suppose $\theta_{K \times M}$ is the weight matrix whose column vector are the mixing weights over $K$ topics. Based on generative model and matrix operations, we can deduce the empirical word-document frequency matrix $X$ and empirical word co-occurrence matrix $Q = XX^T$ are sampled with probability distribution $\beta \theta$ and $\theta \theta^T \beta^T$, respectively. If the assumption is satisfied, it shows the existence of “anchor words” [2] or “novel words” [7] that are unique to each topic, so all columns of $X$ or $Q$ reside in a space generated by a subset of $K$ columns of $X$ or $Q$. Due to the space limitation, we ignore the mathematical derivation.

The RP-GSO algorithm is implemented by Matlab, and the pseudocodes are shown in Fig. 2. In Step 1, we firstly parse each document $d_i$ and get the number of times each word appearing in $d_i$, i.e., $X_i$, then normalize $X_i$ into $\bar{X}_i$, and last we obtain the large matrix $\bar{X}$, which composes of $W$ rows and $M$ columns, each column stands for a document. Then in Step 2, we calculate the word-word co-occurrence matrix $Q$ through matrix operations. Step 3 is very important for our feature selection algorithm, because it employs row $l_2$ normalization to get unit vectors, so that all the vectors are distributed in a unit sphere space, which differs with $l_1$ normalization used in Arora’s algorithm [2], because our algorithm focuses on feature selection, whereas Arora’s algorithm focuses on word-topic matrix recovery, and $l_2$ normalization makes no sense for his method. In Steps 4 and 5, we generate a $W \times T$ random projection matrix $R$ [1] to reduce the dimension of $Q$ (the row-$l_2$ normalized $Q$) from $W \times W$ to $W \times T$, where $T \ll W$. Last, in step 7, we apply the Gram-Schmidt orthogonalization [9] to detect the basis vectors of $Q_{rP}$, in which each column vector may be written as a linear combination of these basis vectors.

The core algorithm to detect basis vectors from a set of vectors, because we need some algorithm to select basis vectors of the set of all row vectors in $Q_{rP}$. As we know, the convex hull of the rows in $Q_{rP}$ will be a simplex where the vertices of this simplex correspond to the “basis features”, if we have infinitely many documents. Since we only have a finite number of documents, the rows of $Q_{rP}$ are only an approximation to their expectation. Here we refer to the definition and theorem in reference [2], we can obtain the following result: There is a combination algorithm that runs in time $O(W^2 + WK/\epsilon^2)$ and outputs a subset of $(d_1, \ldots, d_W)$ of size $K$ that $O(\epsilon/\gamma)$-covers the vertices provided that $20K\epsilon/\gamma^2 < \gamma$, where $\gamma > 0$ and $\epsilon > 0$. The detailed proof can be found in reference [2].

In mathematics, there are many decomposition methods for basis vectors detection, and Gram-Schmidt process is a method for orthonormalising a set of vectors in an inner product space, most commonly the Euclidean space $\mathbb{R}^n$. For vectors $u$ and
Based on the above orthogonalization process, we first detect the farthest point from the origin in \( Q_\text{rp} \) (denoted as \( v_1 \)) as the first basis vector and add its normalized vector \( u_1 \) by vector \( u \) orthogonal to the line spanned by vector \( u \). Therefore, for \( K \) vectors \( \{v_1, v_2, \ldots, v_K\} \), we obtain orthogonal sequence vectors \( \{u_1, u_2, \ldots, u_K\} \) by Eq.2

\[
  u_k = v_k - \sum_{j=1}^{k-1} \text{proj}_u(v_k)
\]

Fig. 2. The Matlab pseudocodes of RP-GSO Algorithm.

### IV. EXPERIMENTAL RESULTS

#### A. Data Sets

**Reuters-21578**

The Reuters corpus is a widely used benchmark collection. According to the ModApte split, we get a collection of 52 categories, containing 6,532 training documents and 2,568 test documents, respectively. Then we filter words whose document frequency is less than 15, and we merge training & testing set into one set with \( M = 9100 \), \( W = 2950, C = 52 \). The number of features in our experiments is set as 20, 50, 100, 200, 500, 1000, 2000, and all \( (3089) \), respectively. The last one is used as comparable baseline.

**20Newsgroup**

The Newsgroup corpus consists of about 20K documents, which are uniformly distributed in 20 categories. We also filter the words with document frequency less than 100, and last we get a corpus with \( M = 19914 \), \( W = 3089, C = 20 \). For 20Newsgroup, the number of features is also set as 20, 50, 100, 200, 500, 1000, 2000, and all \( (3089) \), respectively.

It is worthy noting that we filter some infrequent terms by their document frequency in order to compare with GibbsLDA, which fails if the dimensionality of terms is larger than 4K in our computer environment (Intel Core i5 2.6Ghz and 4GB RAM). However, we observed that our algorithm still obtained best results even if we did not filter these infrequent terms through some comparative experiments, which are not reported in this paper.

#### B. Methodology

We compare our RP-GSO method with 4 unsupervised methods: Fast Conical Hull algorithm, containing two variations based on two exterior point selection criteria (distance and greedy), i.e., FCH(dist) and FCH(greedy) [12]; Fast Anchor Words algorithm (FAW) [2]; and GibbsLDA [10]; Besides, we also compare with 4 supervised feature selection methods: \( \chi^2 \) [16], IG (Information Gain) [16], GS Forward Selection (ForwardGS) [13], and supervised MCFS (superMCFS) [4].
We evaluate our method on two different text mining tasks, i.e., classification and clustering. For classification task, we use LibSVM [5] with linear kernel and 10 fold cross validation to obtain classification error rate [5], [14]. For clustering task, we use kMeans algorithm implemented in Matlab. We evaluate the clustering performance with the average Normalized Mutual Information (NMI) [12] of 5 experiments. Before using LibSVM or kMeans, all the input instances are $l_2$ normalized.

C. Classification Results

Classification error rate on skewed Reuters. Figure 3(a) depicts the classification error rate of LibSVM classifier on Reuters when unsupervised feature selection methods are used. As shown in Figure 3(a), the classification error rate decreases with the increase of the number of features, for all the unsupervised feature selection methods, except GibbsLDA. Our proposed RP-GSO achieves significantly better performance than FAW and FCH with 95% t-test, especially when the number of feature is less than 500, the classification error rate reduction ranges from 10% to 20% comparing against other unsupervised methods. For FCH and FAW methods, the former is worse than the later when the number of features is less than 100; and FCH will perform better when the number of features is bigger than 200. Generally, FCH(dist) performs the second better classification error rate. When the number of features is bigger than 200, FAW method performs worse than others, because the $l_1$ normalization of vectors affects its performance of anchor words detection for classification task. There is an interesting and natural phenomena in GibbsLDA method, our experimental results show that GibbsLDA achieves best when the the number of features is less than 100, even better than our RP-GSO. But its performance reduces drastically when topic number is bigger than 100. The reason is that we use hidden topics as features of document, the supposition is satisfied when the topic number is set small, and the supposition will be destroyed after setting too big topic number. In fact, how to set the hidden topic number in topic model is still an open question.

As shown in Figure 4(b), it demonstrates that our RP-GSO method achieves similar classification error rate comparing against $\chi^2$, IG, and superMCFS supervised feature selection methods, and our RP-GSO method performs slightly better than IG and superMCFS. ForwardGS obtained the worst results. It should be noting that our method does not employ label information, whereas methods else are only effective for labeled corpus.

D. Clustering Results

In this section, we use the average normalized mutual information of 5 random experiments to evaluate the performance of feature selection methods for clustering task. The clustering results demonstrate that our RP-GSO method can achieve competitive NMI score to the well-known supervised $\chi^2$ and IG on skewed Reuters, and better NMI on balanced 20 Newsgroup. Meanwhile, our RP-GSO still performs better than unsupervised feature selection methods for clustering task.

NMI on skewed Reuters. Figure 5(a) shows the comparable results of 5 unsupervised feature selection methods on
skewed Reuters. We can observe that our RP-GSO method still significantly performs better NMI score than others with 95% t-test, except GibbsLDA with less than 100 features. Compared against the second best FCH(dist), the improvement of our method ranges from 4% to 19.9% when the number of features is less than 500. Similar to classification on Reuters, GibbsLDA obtains best performance when the number of features is small, and its performance decreases dramatically with increase of features. For FAW method, its NMI score is significantly lower than our RP-GSO, even lower than FCH methods, the reason may be that clustering task requires more semantically related features, whereas the selected features by FAW cannot satisfy this point.

NMI on balanced Newsgroup. We continue to compare NMI score on balanced 20 Newsgroup. As shown in Figure 6(a), our RP-GSO method performs superiorly in terms of NMI, and the improvement is significant with 95% t-test when features are less than 200. FAW obtains a bad NMI score, which is lower than 0.1 when the number of features is smaller than 1000. It means that FAW method fails in this clustering task. Besides, FCH(dist) method is slightly better than FCH(greedy), and the difference between them is small, so we prefer to FCH(dist), because it is more efficient than FCH(greedy) method.

Figure 6(b) shows that our RP-GSO method achieves competitive NMI score, comparing against $\chi^2$, IG, and superMCFS supervised approaches. When the number of features is less than 200, our RP-GSO is worse than $\chi^2$, but is still better than IG method. Our method performs significantly best if we choose more than 200 features.

E. Parameter Settings
In RP-GSO algorithm, $R$ is a $W \times T$ matrix, and parameter $T$ is manually set. We constructed classification and clustering experiments on both corpora to analyze the performance of RP-GSO with different $T$ values. As shown in Figure 7, the subfigure (a) shows the classification accuracy trend with different $T$ values, and the subfigure (b) shows the NMI trend. We observe that the influence of parameter $T$ is insignificant on both Reuters and 20 News group in terms of classification accuracy and NMI score. Specifically, we set the number of features as 1000 for classification task, and the accuracy fluctuation is very small when $T$ value is from 100 to 3000. The same phenomenon happens for clustering task when the number of features is set as 200. When $T$ is equal to 0, it means no random projection is used. We observe that the accuracy or NMI without random projection is close to that with random projection, but using random projection can significantly reduce computing time.

F. Influence of Input Space
In this section, we observe the influence of input space for our RP-GSO algorithm. Because our RP-GSO algorithm can be applied on any matrix, we directly use the original matrix $X$ as the input of RP-GSO, denoted as RP-GSO-X. Compared to word co-occurrence matrix $Q$, the original word-document matrix $X$ is sparse. The density of $X$ on two data sets are only 1.27% and 1.6%, respectively, which will affect GS-Orthogonalization process.
and Gram-Schmidt process to detect the basis vectors of word co-occurrence matrix. And our extensive experiments on text classification and clustering tasks demonstrate the superiority of our proposed RP-GSO algorithm in terms of classification error rate and normalized mutual information, when LibSVM and kMeans algorithm are used, respectively. Meanwhile, our experiments provide a set of insights on how to achieve good empirical quality for feature selection, that is, $l_2$ normalization and word co-occurrence matrix outperform $l_1$ normalization and document-term matrix. Last, our RP-GSO algorithm can be parallelized on matrix computation, and we will verify the effectiveness and efficiencies of RP-GSO on large scale text corpora in the future work.

VI. ACKNOWLEDGMENTS

This research was supported by the National Natural Science Foundation of China (NSFC) (Grant No 71501003) and China Postdoctoral Science Foundation funded project (Grant No. 2014M550591). The Matlab code of our RP-GSO algorithm can be downloaded from http://www.nlsde.buaa.edu.cn/dqwang.

REFERENCES


