Outlier Detection for Text Data

Ramakrishnan Kannan * Hyenkyun Woo † Charu C. Aggarwal ‡ Haesun Park §

Abstract

The problem of outlier detection is extremely challenging in many domains such as text, in which the attribute values are typically non-negative, and most values are zero. In such cases, it often becomes difficult to separate the outliers from the natural variations in the patterns in the underlying data. In this paper, we present a matrix factorization method, which is naturally able to distinguish the anomalies with the use of low rank approximations of the underlying data. Our iterative algorithm TONMF is based on Block Coordinate Descent (BCD) framework. Our approach has significant advantages over traditional methods for text outlier detection. Finally, we present experimental results illustrating the effectiveness of our method over competing methods.

1 Introduction

The problem of outlier detection is that of finding data points which are unusually different from the rest of the data set. Such outliers are also variously referred to as anomalies, deviants, discordants or abnormalities in the data. Since outliers correspond to unusual observations, they are often of interest to the analyst in finding interesting anomalies in the underlying generating process. The problem of outlier analysis is applicable to a wide variety of domains such as machine monitoring, financial markets, environmental modeling and social network analysis. Correspondingly, the problem has been studied in the context of different data types which arise in these domains, such as multidimensional data, spatial data, and discrete sequences. Numerous books and surveys have been written on the problem of outlier detection [1, 6].

In this paper, we will study the problem of text outlier analysis. The problem of text outlier analysis has become increasingly important because of the greater prevalence of web-centric and social media applications, which are rich in text data. Some important applications of text outlier analysis are as follows:

- *Web Site Management:* An unusual page from a set of articles in a web site may be flagged as an outlier. The knowledge of such outliers may be used for web site management.
- *Sparse High Dimensional Data:* While the methods discussed in this paper have text applications in mind, they can be used for other sparse high dimensional domains. For example, such methods can be used for market basket data sets. Unusual transactions may sometimes provide an idea of fraudulent behavior.
- *News Article Management:* It is often desirable to determine unusual news article from a collection of news documents. An unusual news from a group of articles may be flagged as an interesting outlier.

While text is an extremely important domain from the perspective of outlier analysis, there are surprisingly few methods which are *specifically focused* on this domain, even though many generic methods such as distance-based methods can be easily adapted to this domain [14, 21], and are often used for text outlier analysis. Domains such as text are particularly challenging for the problem of outlier analysis, because of their sparse high dimensional nature, in which only a small fraction of the words take on non-zero values. Furthermore, many words in a document may be topically irrelevant to the context of the document and add to the noise in the distance computations. For example, the word "*Jaguar*" may correspond to a car, or a cat depending on the context of the document. In particular, the significance of a word can be interpreted only in terms of the structure of the data within the context of a particular data locality. As a result, document-to-document similarity measures often lose their robustness. Thus, commonly used outlier analysis methods for multidimensional data, such as distance-based methods, are not particularly effective for text data. Our experiments also validate this observation.

In this paper, we will use non-negative matrix factorization (NMF) methods to address the aforementioned challenges in text anomaly detection. One advantage of matrix factorization methods is that they decompose the term-document structure of the underlying corpus into a set of semantic term clusters and document clusters. The semantic nature of this decomposition provides the context in which a document may be interpreted for outlier analysis. Thus, documents can be decomposed into word clusters, and words are decomposed into document clusters with

*Oak Ridge National Laboratory,kannanr@ornl.gov
†Korea University of Technology and Education,hyenkyun@koreatech.ac.kr
‡IBM T. J. Watson Research Center,charu@us.ibm.com
§Georgia Institute of Technology,hpark@cc.gatech.edu

This manuscript has been authored by UT-Battelle, LLC under Contract No. DE-AC05-00OR22725 with the U.S. Department of Energy. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non- exclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes. The Department of Energy will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan (http://energy.gov/downloads/doe-public-access-plan).

a low-rank[1] approximation. Outliers are therefore defined as data points which cannot be naturally expressed in terms of this decomposition. By using carefully chosen model formulations, one can further sharpen the matrix-factorization method to reveal document-centric outliers. One challenge in this case, is the design of a matrix factorization approach, that is optimized to anomaly detection, results in a non-standard formulation. and we will carefully design an optimization method based on Block Coordinate Descent (BCD) to solve this problem. The NMF model also has the advantage of providing better interpretability, and it can also provide insights into why a document should be considered an outlier. We present extensive experimental results on many data sets, and compare against a variety of baseline methods. We show significant improvements achieved by the approach over a variety of other methods.

This paper is organized as follows. The remainder of this section discusses the related work. Section 2 introduces the model for outlier analysis. The algorithm to solve this model is provided in section 3. Section 4 provides the experimental results. The conclusions and summary are contained in section 5. Our code can be downloaded from `https://github.com/ramkikannan/outliernmf` and tried with any text dataset. For the convenience of the readers, we provide an extended version of the paper with supplement information in [11].

1.1 Related Work The outlier analysis problem has been studied extensively in the literature [1, 6]. Numerous algorithms have been proposed in the literature for outlier detection of conventional multidimensional data [2, 4, 14, 21]. The key methods, which are used frequently for outlier analysis include distance-based methods [14, 21], density-based methods [4], and subspace methods [2, 12, 17, 20, 16]. In distance-based methods, data points are declared outliers, when they are situated far away from the dense regions in the underlying data. Typically, indexing or other summarization schemes may be used in order to improve the efficiency of the approach. In density-based methods [4], data points with low local density with respect to the remaining points are declared outliers. In addition, a number of subspace methods [2, 12, 17, 20, 16] have been proposed recently, in which outliers are defined on the basis of subspace behavior of the underlying data.

Most of the traditional multidimensional methods [6, 1] can also be extended to text data, though they are not particularly suited to the latter. Some methods have been designed for outlier detection with matrix factorization in network data sets [23], that are not applicable to text data. Text data is uniquely difficult because of its sparse and high dimensional nature. As a result, many of the outliers detected using conventional methods may simply correspond to noisy text segments. Therefore, careful modeling is required with the use of matrix factorization methods.

[1]In this paper, we use the terms "low rank approximation" and "matrix factorization" interchangeably. Similarly, we used the terms "anomalies" and "outliers" interchangeably.

Over the last decade, Non-negative Matrix Factorization (NMF) has emerged as another important low rank approximation technique, where the low-rank factor matrices are constrained to have only non-negative elements. Lee and Seung [18] introduced a multiplicative update based low rank approximation with non-negative factors to overcome the challenges of truncated SVD. Subsequent to this work, NMF has received enormous attention and has been successfully applied to a broad range of important problems in areas including computer vision, community detection in social networks, visualization, recommender systems bioinformatics, etc. In spite of broad range of applications, NMF's literature in text domain is scarce. Xu *et. al.* [26] experimented with NMF for document clustering instead of SVD based Latent Semantic Indexing (LSI). Other than applications of NMF in the text domain, Gaussier and Goutte [10] established the equivalence between NMF and pLSA. Similarly, Ding *et. al.* [7] explained the equivalence between NMF and pLSI.

In this paper, we use an NMF approach for concise modelling of the patterns, the background, and the anomalies in the underlying data. It should be pointed out that NMF is similar to the generative models of text such as pLSI and LDA [10] [7] [22], though NMF often provides better interpretability. Our important challenge is to model the outliers along with the low rank space of the input matrix. We identified $\ell_{1,2}$-norm as an appropriate approach for factorization in outlier analysis. Recently, the researchers have used $\ell_{2,1}$-norm in their models to solve various problems, though the corresponding solution techniques are not easily generalizable to the $\ell_{1,2}$-norm. Yang et.al., [27], under the assumption that the class label of input data can be predicted by a linear classifier, incorporate discriminative analysis and $\ell_{2,1}$-norm minimization into a joint framework for unsupervised feature selection problem. Similarly, Liu *et al* [19], solve $\ell_{2,1}$-norm regularized regression model for joint feature selection from multiple tasks. They also propose to use Nesterov's method to solve the optimization problem with non-smooth $\ell_{2,1}$-norm regularization. Also, Kong *et al* [15] propose a robust formulation of NMF using $\ell_{2,1}$-norm loss function for data with noises.

1.2 Our Contributions Text data is uniquely challenging to outlier detection both because of its sparsity and high dimensional nature. Given the relevant literature for NMF and text outliers, we propose the first approach to detect outliers in text data using non-negative matrix factorization. We extend the fact that NMF is similar to pLSI and LDA generative models and model the outliers using the $\ell_{1,2}$-norm. This particular formulation of NMF is non-standard, and requires careful design of optimization methods to solve the problem. We solve the resulting optimization problem using block coordinate descent technique. We also present extensive experimental results both on text and other kinds of market basket data sets. We show significant improvements achieved by the approach over other baseline methods.

Notation	Explanation
$\mathbf{A}=[\mathbf{a}_1 \cdots \mathbf{a}_n] \in \mathbb{R}_+^{m \times n}$	Document-word matrix
m	Vocabulary size
n	Number of documents
$\mathbf{Z} \in \mathbb{R}^{m \times n}$	Outlier matrix
$r < rank(\mathbf{A})$	Rank
$\mathbf{W} \in \mathbb{R}_+^{m \times r}$	Term-Topic matrix
$\mathbf{H} \in \mathbb{R}_+^{r \times n}$	Topic-Document matrix
$\mathbf{A}^{(i)}$	Matrix $\mathbf{A}$ from the i^{th} iteration
$\|\mathbf{A}\|_{1,2}$	$\sum_{i=1}^{n} \|\mathbf{a}_i\|_{\ell_2}$ ℓ_{12}-Norm where, $\mathbf{a}_i \in \mathbb{R}^m$ is the i-th column of $\mathbf{A}$

Table 1: Notations used in the paper

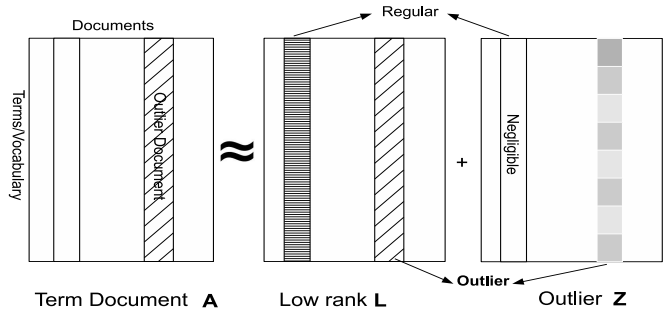

Figure 1: Text Outliers Using NMF

2 Matrix Factorization Model

This section will present the matrix factorization model which is used for outlier detection.

For the reader's convenience, the notations used in the paper are summarized in Table 1. Let $\mathbf{A}$ be the matrix representing the underlying data. In the context of a text collection, this corresponds to a term-document matrix, where terms correspond to rows and documents correspond to columns. In other words, a_{ij} denotes the number of times the term i appears in document j. Generally, we can write $\mathbf{A}$ as follows:

$$\mathbf{A} = \mathbf{L_0} + \mathbf{Z_0}. \tag{2.1}$$

Here, $\mathbf{L_0}$ is a low rank matrix and $\mathbf{Z_0}$ represents the matrix of outlier entries. Typically, the matrix $\mathbf{L_0}$ represents the documents created by a lower rank generative process (such as that modeled by pLSI), and the parts of the documents that do not correspond to the generative process are represented as part of the matrix $\mathbf{Z_0}$. In real world scenarios, the outlier matrix $\mathbf{Z_0}$ contains entries which are very close to zero, and only a small number of entries have *significantly* non-zero values. These significantly nonzero entries are often present in only a small fraction of the columns. Columns which are fully representable in terms of factors are consistent with the low rank behavior of the data, and therefore *not* outliers. The rank of $\mathbf{L}_0$ is not known in advance, and it can be expressed in terms of its underlying factors.

$$\mathbf{L}_0 \approx \mathbf{W}_0 \mathbf{H}_0$$

Here, the two matrices have dimensions $\mathbf{W}_0 \in \mathbb{R}_+^{m \times r}$, $\mathbf{H}_0 \in \mathbb{R}_+^{r \times n}$, and $r \leq rank(\mathbf{L}_0)$. The matrices $\mathbf{W}_0$ and $\mathbf{H}_0$ are non-negative, and this provides interpretability in terms of being able to express a document as a non-negative linear combination of the relevant basis vectors, each of which in itself can be considered a frequency-annotated bag of words (topics) because of its non-negativity. Specifically, $\mathbf{H_0}$ corresponds to the coefficients for the basis matrix $\mathbf{W_0}$. Intuitively, this corresponds to the case that every document $\mathbf{a}_i$, is represented as the linear combination of the r topics. In cases, where this is *not* true, the document is an outlier, and those unrepresentable sections of the matrix are captured by the non-zero entries in the $\mathbf{Z_0}$ matrix. In real scenarios, the entries in this matrix are often extremely skewed, and the small number of non-zero entries very obviously expose the outliers. The decomposition of the matrix into different component is pictorially illustrated in Figure 1.

In order to determine the best low rank factorization, one must try to optimize the aggregate values of the residuals in the matrix. This can of course be done in a variety of ways, depending upon the goals of the underlying factorization process. We model the determination of the matrices $\mathbf{W}$, $\mathbf{H}$, and $\mathbf{Z}$, as the following optimization problem:

$$(\mathbf{W}_0, \mathbf{H}_0; \mathbf{Z}_0) = \underset{\mathbf{W} \geq 0, \mathbf{H} \geq 0; \mathbf{Z}}{\operatorname{argmin}} \frac{1}{2} \|\mathbf{A} - \mathbf{W}\mathbf{H} - \mathbf{Z}\|_F^2 + \alpha \|\mathbf{Z}\|_{1,2} \tag{2.2}$$

The specific location of outliers in each column does not have a closed form solution, since the $\ell_{1,2}$-norm penalty is applied to $\mathbf{Z}$. The logic for applying the $\ell_{1,2}$-norm in the context of the outlier detection problem is as follows. Each entry in the $\mathbf{Z}$ corresponds to a term in a document, whereas we are interested in the outlier behavior of entire document. This aggregate outlier behavior of the document x can be modeled with the ℓ_2 norm score of a particular column $\mathbf{z}_x$. In a real scenario, if a large segment of a document x is not representable as the linear combination of the r topics through $\mathbf{L_0}$, the corresponding column $\mathbf{z}_x$ in the matrix $\mathbf{Z}$ will be compensated by having more entries in its column. In other words, we will have a higher ℓ_2 value for the corresponding column $\mathbf{z}_x$, and this corresponds to a higher outlier score. Furthermore, the $\ell_{1,2}$-norm penalty on $\mathbf{Z}$ defines the sum of the ℓ_2 norm outlier scores over all the documents. Therefore, the optimization problem essentially tries to find the best model, an important component of which is to minimize the sum of the outlier scores over all documents. While a variety of different (and more commonly used) penalties such as the Frobenius norm are available for matrix factorization models, we have chosen the $\ell_{1,2}$-norm penalty because of its intuitive significance in the context of the outlier detection problem, and its tendency to create skewed outlier scores across the columns of the matrix. As we will see in the next section, this comes at the expense of a formulation which is more difficult to solve algorithmically.

For high dimensional data, sparse coefficients are desirable for obtaining an interpretable low rank matrix $\mathbf{WH}$. For this

purpose, we add the ℓ_1-penalty on $\mathbf{H}$:

$$\min_{\mathbf{W}\geq 0,\mathbf{H}\geq 0;\mathbf{Z}} \frac{1}{2}\|\mathbf{A}-\mathbf{W}\mathbf{H}-\mathbf{Z}\|_F^2+\alpha\|\mathbf{Z}\|_{1,2}+\beta\|\mathbf{H}\|_1 \quad (2.3)$$

The constant α defines the weight for the outlier matrix $\mathbf{Z}$ over the recovery of the low rank space $\mathbf{L}$ and the sparsity term. In the case of outlier detection in text documents, we give more weight for the outlier matrix over the low rank representation $\mathbf{L}$. This problem does not have a closed form solution, and therefore we cannot directly recover the low rank matrix $\mathbf{WH}$ in closed form. However, we can recover the column space. Without non-negativity constraints, this property is also known as the rotational invariant property [8, 25]. This particular formulation of the matrix factorization model is a bit different from the commonly used formulations, and off-the-shelf solutions do not exist for this scenario. Therefore, in a later section, we will carefully design an algorithm with the use of block coordinate descent for this problem.

In order to understand the modeling of the outliers better, we present the readers with a toy example from a real world data set, to show how skewed the typical values of the corresponding column $\mathbf{z}_x$ may be in real scenarios. In this case, we used the *BBC* dataset[2]. This dataset consists of documents from BBC news website corresponding to stories in area of business, entertainment, politics, sport, tech from 2004-2005 . We took all the documents from business and politics and 50 documents from tech labeled as outliers. We randomly permuted the columns to shuffle the outliers in the matrix to avoid any spatial bias. We computed the $\mathbf{Z}$ matrix and generated the ℓ_2 scores of the columns of outlier matrix $\mathbf{Z}$. Figure 2 shows the outlier(ℓ_2) scores of the documents. The X-axis illustrates the index of the document, and the Y-axis illustrates the outlier score. It is evident that the scores for some columns are so close to zero, that they cannot even be seen on the diagram drawn to scale. These columns also happened to be the non-outlier/regular documents of the collection. Such documents $\mathbf{a}_x \in \mathbb{R}^m$ correspond to the low rank space, and are approximately representable as a product of the basis matrix $\mathbf{W}$ with the corresponding column vector of coefficients $\mathbf{h}_x \in \mathbb{R}^r$ drawn from $\mathbf{H}$. However, the documents that are not representable in such a low rank space have a large outlier score. From the distribution of the outlier score, we can also observe that the scores of outlier documents against non-outliers are clearly separable, by using a simple statistical mean and standard deviation analysis. Therefore, while we use the scores to rank the documents in terms of their outlier behavior, the skew in the entries ensures that it is often easy to choose a cut-off in order to distinguish the outliers from the non-outliers.

In the following sections, we will analyze the property and performance of the model (2.3) for outlier detection problems.

3 Algorithmic Solution

As discussed earlier our technique is based on NMF, the particular formulation (2.3), which is suited to outlier analysis, is relatively uncommon, and does not have a closed form solution. In order to address this issue we use a Block Coordinate Descent (BCD) framework and its application to solve the optimization problem (2.3). The BCD framework is a popular choice not only because of the ease in implementation, but also because it is scalable. Kim, He and Park [13] provide the general introduction to BCD technique and its application to NMF in detail. We will relate the BCD framework to our non-negative matrix factorization problem, and explain our algorithm Text Outliers using Nonnegative Matrix Factorization(TONMF) in detail.

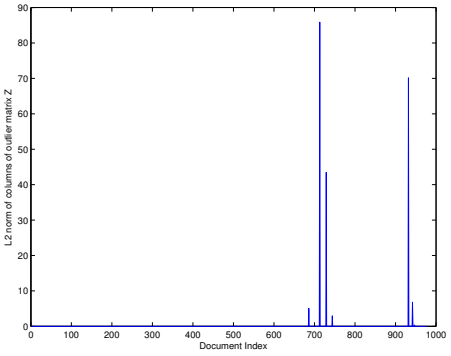

Figure 2: ℓ_2 norm of columns of Z outlier matrix

3.1 Text Outliers using Nonnegative Matrix Factorization (TONMF)

In this section, we propose an efficient algorithm for the outlier detection model (2.3).

To determine the $\mathbf{Z},\mathbf{W},\mathbf{H}$ for the aforementioned optimization problem (2.3), we use the block coordinate descent method. In other words, by fixing $\mathbf{W},\mathbf{H}$, we determine the optimal $\mathbf{Z}$ as vector blocks $\mathbf{z_1},\cdots,\mathbf{z_n}$ and vice versa. Due to $\ell_{1,2}$-norm, this optimization corresponds to the two block non-smooth BCD framework.

$$\begin{aligned}\mathbf{Z}^{(k+1)} &\leftarrow \operatorname*{argmin}_{\mathbf{Z}} \frac{1}{2}\|\mathbf{A}-\mathbf{Z}-\mathbf{W}^{(k)}\mathbf{H}^{(k)}\|_F^2+\alpha\|\mathbf{Z}\|_{1,2}\\ (\mathbf{W}^{(k+1)},\mathbf{H}^{(k+1)}) &\leftarrow \operatorname*{argmin}_{\mathbf{W}\geq\mathbf{0},\mathbf{H}\geq\mathbf{0}} \frac{1}{2}\|\mathbf{A}-\mathbf{W}\mathbf{H}-\mathbf{Z}^{(k+1)}\|+\beta\|\mathbf{H}\|_1\end{aligned} \quad (3.4)$$

Regarding $\mathbf{Z}=[\mathbf{z}_1,\ldots,\mathbf{z}_n]$, the minimization problem in (3.4) has a separable structure:

$$\mathbf{Z}^{(k+1)}=\operatorname*{argmin}_{\mathbf{Z}} \sum_i \frac{1}{2}\|\bar{\mathbf{a}}_i-\mathbf{z}_i\|_2^2+\alpha\|\mathbf{z}_i\|_2$$

where $\bar{\mathbf{a}}_i=\mathbf{a}_i-(\mathbf{W}^{(k)}\mathbf{H}^{(k)})_i$. Therefore, we only need to define a solution with respect to one variable $\mathbf{z}_i$. Thus, we partition the matrix $\mathbf{Z}$ into vector blocks $\mathbf{z}_i$ and construct $\mathbf{Z}$ as a set of vectors $\mathbf{z}_i$. Also, the blocks $\mathbf{z}_i$ is independent of $\mathbf{z}_j,\forall i\neq j$. That is, the closed form solution of $\mathbf{z}_i$ is dependent only on $\bar{\mathbf{a}}_i$. When all other blocks of $\mathbf{w_1},\cdots,\mathbf{w_r},\mathbf{h_1},\cdots,\mathbf{h_r}$, are fixed, every vector $\mathbf{z}_i \in \mathbf{Z}$, can be solved to optimal in parallel. Thus, we adhere to

[2] http://mlg.ucd.ie/datasets/bbc.html

BCD framework of solving the vector blocks of $\mathbf{z}_i$, to optimal, when all the other blocks are fixed.

THEOREM 3.1. *The solution of the following minimization problem*

$$\mathbf{z}_i^* = \underset{\mathbf{z}_i}{\operatorname{argmin}} f(\mathbf{z}_i) = \frac{\gamma}{2}\|\mathbf{z}_i - \mathbf{a}_i\|_2^2 + \alpha\|\mathbf{z}_i\|_2$$

is the generalized shrinkage operator:

$$\mathbf{z}_i^* = \mathit{shrink}(\mathbf{a}_i, \frac{\alpha}{\gamma})$$

defined as:

$$\mathit{shrink}(\mathbf{a}_i, C) = \mathit{max}(\|\mathbf{a}_i\|_2 - C, 0)\frac{\mathbf{a}_i}{\|\mathbf{a}_i\|_2}$$

Proof.

$$\frac{\partial f(\mathbf{z}_i)}{\partial \mathbf{z}_i} = \gamma(\mathbf{z}_i - \mathbf{a}_i) + \alpha\frac{\mathbf{z}_i}{\|\mathbf{z}_i\|}$$

When $\|\mathbf{a}_i\|_2 \le \frac{\alpha}{\gamma}$,

$$f(\mathbf{z}_i) \ge \frac{\gamma}{2}(\|\mathbf{z}_i\|_2^2 + \|\mathbf{a}_i\|_2^2) + (\alpha - \gamma\|\mathbf{a}_i\|_2)\|\mathbf{z}_i\|_2$$

Therefore we have:

$$\underset{\mathbf{z}_i}{\operatorname{argmin}} f(\mathbf{z}_i) = 0.$$

When $\|\mathbf{a}_i\|_2 \ge \frac{\alpha}{\gamma}$, let $\mathbf{z}_i = c\mathbf{a}_i$ then

$$\frac{\partial f(\mathbf{z}_i)}{\partial \mathbf{z}_i} = \gamma(\mathbf{z}_i - \mathbf{a}_i) + \alpha\frac{\mathbf{z}_i}{\|\mathbf{z}_i\|_2} = [\gamma(c-1) + \frac{\alpha}{\|\mathbf{a}_i\|_2}]\mathbf{a}_i = 0$$

where

$$c = 1 - \frac{\alpha}{\gamma}\frac{1}{\|\mathbf{a}_i\|_2}.$$

Therefore, we get

$$\mathbf{z}_i = (\|\mathbf{a}_i\|_2 - \frac{\alpha}{\gamma})\frac{\mathbf{a}_i}{\|\mathbf{a}_i\|_2}$$

Now, utilizing the generalized shrinkage operator as defined in [9][24],

$$\mathbf{z}_i^* = \text{shrink}(\mathbf{a}_i, C) = \max(\|\mathbf{a}_i\|_2 - C, 0)\frac{\mathbf{a}_i}{\|\mathbf{a}_i\|_2}$$

where $C = \alpha/\gamma$. ∎

Now, we need to solve the following NMF model with sparsity constraints on $\mathbf{H}$:

$$(\mathbf{W}^{(k+1)}, \mathbf{H}^{(k+1)}) = \underset{\mathbf{W}\ge 0, \mathbf{H}\ge 0}{\operatorname{argmin}} \|\bar{\mathbf{A}} - \mathbf{W}\mathbf{H}\|_F^2 + \beta\|\mathbf{H}\|_1$$

where $\bar{\mathbf{A}} = \mathbf{A} - \mathbf{Z}^{(k+1)}$. Let

$$\mathcal{F}(\mathbf{w}_1,...,\mathbf{w}_r; \mathbf{h}_1,...,\mathbf{h}_r) = \|\bar{\mathbf{A}} - \sum_{i=1}^{r}\mathbf{w}_i\mathbf{h}_i\|_F^2 + g(\mathbf{h}_1,...,\mathbf{h}_r). \tag{3.5}$$

where $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_r]$ and $\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_r]^T$. For any $j \in \{1,...,r\}$, (3.5) can be rewritten as

$$\|\bar{\mathbf{A}} - \sum_{i=1}^{r}\mathbf{w}_i\mathbf{h}_i^T\|_F^2 = \|\bar{\mathbf{A}} - \sum_{i=1, i\ne j}^{r}\mathbf{w}_i\mathbf{h}_i^T - \mathbf{w}_j\mathbf{h}_j^T\|_F^2. \tag{3.6}$$

The following is the framework of the block coordinate descent method with a separable regularizer such as the Frobenius norm. We iteratively minimize $\mathcal{F}(\mathbf{W},\mathbf{H})$ with respect to each column of $\mathbf{W}$ and $\mathbf{H}$:

$$\begin{cases} \text{for } j = 1...r \\ \quad \mathbf{h}_j^{(k+1)} = \underset{\mathbf{h}_j \ge 0}{\operatorname{argmin}} \frac{\alpha}{2}\|\mathbf{w}_j^{(k)}\mathbf{h}_j^T - (\bar{\mathbf{A}} - \tilde{\mathbf{W}}_j^{(k)})\|_F^2 \\ \qquad\qquad + g(\mathbf{h}_1^{(k+1)},...,\mathbf{h}_j,...,\mathbf{h}_r^{(k)}) \\ \text{end} \\ \text{for } j = 1...r \\ \quad \mathbf{w}_j^{(k+1)} = \underset{\mathbf{w}_j \ge 0}{\operatorname{argmin}} \|\mathbf{w}_j(\mathbf{h}_j^{(k+1)})^T - (\bar{\mathbf{A}} - \tilde{\mathbf{H}}_j^{(k+1)})\|_F^2 \\ \text{end} \end{cases} \tag{3.7}$$

where

$$\tilde{\mathbf{W}}_j^{(k)} = \sum_{i=1}^{j-1}\mathbf{w}_i^{(k)}(\mathbf{h}_i^{(k+1)})^T + \sum_{i=j+1}^{r}\mathbf{w}_i^{(k)}(\mathbf{h}_i^{(k)})^T,$$

and

$$\tilde{\mathbf{H}}_j^{(k+1)} = \sum_{i=1}^{j-1}\mathbf{w}_i^{(k+1)}(\mathbf{h}_i^{(k+1)})^T + \sum_{i=j+1}^{r}\mathbf{w}_i^{(k)}(\mathbf{h}_i^{(k+1)})^T.$$

According to Theorem 3.1, the solution of $\mathbf{z}_i$ is independent of $\mathbf{z}_j, \forall i \ne j$, and it enables us to solve the solution in parallel. This is very useful when computing for very large input matrices. Similarly, the vector blocks of $\mathbf{W},\mathbf{H}$ can also be updated in parallel. Now, we have all the building blocks for the Text Outliers using Nonnegative Matrix Factorization algorithm. We will be using Theorem 3.1 and the update for $\mathbf{W},\mathbf{H}$ from equation (3.7). The Algorithm 1, gives the outline of the TONMF. The real implementation of the algorithm can be downloaded from `https://github.com/ramkikannan/outliernmf` and tried with any text dataset.

4 Experimental Results

In this section, we present the experiments on text outlier analysis using matrix factorization. We used both real and synthetic data sets to test our algorithm. The real data sets correspond to the well known *RCV20*, *Reuters* and *Wiki People* data, whereas the synthetic data set was created using a well known market basket generator described later. It should be pointed out that these data sets were not originally designed for outlier analysis, and they have no ground truth information available. Therefore, some additional pre-processing is needed to be applied to the real data sets, in order to isolate ground truth classes, and use them effectively for the outlier analysis problem. In this section, we will describe the data sets, their preparation, the performance criteria and the results obtained by our algorithm. At the end of this section, we will also present a discussion that provides interesting insights about the effectiveness of algorithm TONMF .

4.1 Data Sets The experiments were conducted with both labelled real and synthetic data sets. For all the datasets described below, we randomly permuted the positions of the outliers and regular data points in the matrix representation, to avoid any

Algorithm 1: Text Outliers using Nonnegative Matrix Factorization (TONMF)

input : Matrix $\mathbf{A} \in \mathbb{R}_+^{m \times n}$, reduced rank r, α, β
output Matrix $\mathbf{W} \in \mathbb{R}_+^{m \times r}$, $\mathbf{H} \in \mathbb{R}_+^{r \times n}$, $\mathbf{Z} \in \mathbb{R}^{m \times n}$
:
// Rand initialization of **W, H, Z**
Initialize ***W, H, Z*** *as a nonnegative random matrix* ;
while *stopping criteria* $\mathfrak{C}_1$ *not met* **do**
  // Compute Z for the given $\mathbf{A}, \mathbf{W}, \mathbf{H}, \alpha, \beta$ based on Theorem 3.1
  for $i \leftarrow 1$ **to** n **do**
    $\mathbf{z_i} \leftarrow max(\|\mathbf{a}_i\|_2 - \frac{\alpha}{\gamma}, 0) \frac{\mathbf{a}_i}{\|\mathbf{a}_i\|_2}$
  while *stopping criteria* $\mathfrak{C}_2$ *not met* **do**
    for $j \leftarrow 1$ **to** r **do**
      $\mathbf{h}_j^{(k+1)} = \underset{\mathbf{h}_j \geq 0}{\operatorname{argmin}} \frac{\alpha}{2} \|\mathbf{w}_j^{(k)} \mathbf{h}_j^T - (\bar{\mathbf{A}} - \tilde{\mathbf{W}}_j^{(k)})\|_F^2 + g(\mathbf{h}_1^{(k+1)}, \cdots, \mathbf{h}_j, \cdots, \mathbf{h}_r^{(k)})$;
    for $j \leftarrow 1$ **to** r **do**
      $\mathbf{w}_j^{(k+1)} = \underset{\mathbf{w}_j \geq 0}{\operatorname{argmin}} \|\mathbf{w}_j (\mathbf{h}_j^{(k+1)})^T - (\bar{\mathbf{A}} - \tilde{\mathbf{H}}_j^{(k+1)})\|_F^2$;

unforeseen bias in the algorithm.

RCV20 Data Set: [3] We took all data points from two randomly chosen classes - *IBM* and *Mac Hardware*. In addition, 50 data points from a randomly chosen class - *Windows Operating System (OS)*. As it turns out, this is a rather hard problem for our algorithm because of some level of relationship between one of the rare classes and the base data. Specifically, *Windows Operating System* and *IBM Hardware* are both computer related subjects, and the former is often used with the latter.

Reuters-21578 Data Set: [4] We selected those documents that belong to only one category. There were totally 5768 documents that belong to the category *earn* and *acq*. The outliers were 100 documents from category *interest*.

Wiki People Dataset: This is a subset of the dataset collected by Blasiak et.al., [3]. The dataset is constructed by crawling Wikipedia starting from `http://en.wikipedia.org/wiki/Category:Lists_of_politicians` to a depth of four. Pages describing people were extracted from the list of all crawled pages. Text from the body paragraphs of the pages were extracted, and section headings were used as labels for blocks of text. Text blocks were assumed to begin with $<p>$ and end with $</p>$. Only text in section headings that occurred 10 times or more was retained. Words were stemmed, stopwords were removed, and words of length at least 3 and at most 15 were considered. The words need to occur at least 4 times in at least 2 documents to be considered important enough to be retained. From the collected data, the sections *Career* and *Life* were chosen as non-outlier and whereas the small section *Death* was chosen as outlier.

Market Basket Data Generator: For large synthetic sparse matrices that is similar to text data, we used IBM Synthetic Data Generation Code for Associations and Sequential Patterns – market-basket data generator, that is packaged as part of *Illimine*[5] software. We set the average length of the transaction to be 300 and number of different items to be 50,000. Note that this generator uses a random seed, and by changing the seed, it is possible to completely change the transaction distribution, even if all other parameters remain the same. We generated 10,000 data points as a group of four different sets of 2500 data points with randomly chosen seed values. In addition, the rare class contained 250 data points from a single seed value.

4.2 Performance Metrics The effectiveness was measured in terms of the Receiver Operating Characteristics (ROC) curve drawn on the outlier scores. We use the area under the ROC curve – the defacto metric for evaluation in outlier analysis. The idea of this curve is to evaluate a *ranking* of outlier scores, by examining the trade-off between the true positives and false positives, as the threshold on the outlier score is varied in a range. By using different thresholds, it is possible to obtain a relatively larger or smaller number of true positives with respect to the false positives.

Let $S(t)$ be the set of outliers determined by using a threshold t on the outlier scores. In this case, the *True Positive Rate* is graphed against the *False Positive Rate*. The true positive rate $TPR(t)$ is defined in the same way as the metric of recall is defined in the IR literature. The false positive rate $FPR(t)$ is the percentage of the falsely reported positives out of the ground-truth negatives. Therefore, for a data set D with ground truth positives G, these definitions are as follows:

$$TPR(t) = Recall(t) = 100 * \frac{|S(t) \cap G|}{|G|}$$

$$FPR(t) = 100 * \frac{|S(t) - G|}{|D - G|}$$

Note that the end points of the ROC curve are always at $(0,0)$ and $(100,100)$, and a random method is expected to exhibit performance along the diagonal line connecting these points. The *lift* obtained above this diagonal line provides an idea of the accuracy of the approach. The area under the ROC curve provides a measure of the accuracy. A random algorithm would have an area of 0.5 under the ROC curve. The ROC curve was used to provide detailed insights into the trade-offs associated with the method, whereas the area under the ROC curve was used in order to provide a summary of the performance of the method.

4.3 Baseline Algorithms The baselines used by our approach were as follows:

[3] `http://qwone.com/~jason/20Newsgroups/`

[4] `http://archive.ics.uci.edu/ml/datasets/Reuters-21578+Text+Categorization+Collection`

[5] `http://illimine.cs.uiuc.edu/`

Distance-based Algorithm: The first baseline is the classical distance-based algorithm frequently used for outlier detection [14, 21]. The outliers were ranked based on distances in order to create an ROC curve, rather than using a specific threshold as in [14]. In addition, we gave the k-nearest neighbour algorithm an advantage by picking a value of k optimally based on area under ROC curve by sweeping k from 1 to 50. Note that such an advantage would not be available to the baseline under real scenarios, since the ground-truth outliers in the data are unknown, and therefore the ROC curve cannot be optimized.

Simplified Low Rank Approximation: We used a low rank approximation based on Singular Value Decomposition (SVD). For a given matrix $\mathbf{A}$, a best r-rank approximation $\hat{\mathbf{A}}_r$ is given by $\hat{\mathbf{A}}_r = \mathbf{U}\mathbf{S_r}\mathbf{V}^\intercal$, where $\mathbf{S_r} = diag(\sigma_1,\cdots,\sigma_r,0,\cdots,0)$. That is, the trailing $rank(\mathbf{A})-r$ in the descending ordered singular values are set to 0. It is natural to understand that the outlier documents require linear combination of many basis vectors. Thus the ℓ_2 norm on the $\sqrt{S_r}V^\intercal$ can be used a score to determine the outliers. In the graphs, we use SVD as the legend to represent this baseline. For the SVD approach, we used the same low rank as our algorithm.

Robust Principal Component Analysis (RPCA) : Recently Candes et.al.,[5], proposed a new technique called Robust PCA that is insensitive to noises and outliers. It is important to note that both PCA and NMF are different forms of low rank approximation. Hence, we wanted to leverage the output of RPCA and recover the outliers. RPCA yields two matrices (1) a low rank matrix - $\mathbf{L}$ and (2) a sparse matrix $\mathbf{S}$ such that $\mathbf{A} \approx \mathbf{L}+\mathbf{S}$, where $\mathbf{A}$ is the given input matrix. The main disadvantage of RPCA is its larger memory requirements. Retaining $\mathbf{L}$,$\mathbf{S}$ for large matrices require significant memory. We used the ℓ_2 norm on the $\mathbf{S}$ as an outlier score for every document. In the graphs, we use $RPCA$ as the legend to represent this baseline.

4.4 Effectiveness Results We first present the ROC curves for the different data sets. The ROC curve for the *Reuters* dataset is illustrated in Figure 3. In this case, our algorithm shows a drastic improvement over both the baseline algorithms. This is evident from the rather large lift in the chart. The area under the ROC for Our algorithm TONMF was 0.9340. The k-NN approach performed quite poorly, and had an area under the ROC curve of 0.5370. This is slightly better than random performance. The area under ROC for the SVD method was 0.5816 and $RPCA$ was 0.6120, which is better than the k-NN method, but still significantly less than the proposed algorithm.

The comparison of our algorithm with baselines for the *RCV20* data set is shown in Figure 5. As discussed in the data generation section, this is a particularly challenging data set, because of the similarity in the vocabulary distribution between the rare class, and the regular class. It is evident that our algorithm TONMF performed better than the SVD, $RPCA$ and the k-NN method. However, the lift in the ROC curve for all the methods is not particularly significant, because of the inherently challenging nature of the data set. The k-NN method performed particularly poorly in this case. In a later section, we will provide some insights about the fact that some of this "poor" performance is because of the noise in the data set itself, where some of the points in the regular class should really be considered outliers. We generated a datasets in RCV20 where we just changed the outlier class to *christian religion*. We received a best ROC of 0.9732 and it is not shown in Figure 5.

Figure 9 shows the comparison of our algorithm $TONMF$ against the baselines for the *Wiki People* data set. The area under the ROC for k-NN was 0.5395, which is rather poor. All the other methods performed better than k-NN with area under the ROC for SVD being 0.5670 and $RPCA$ being 0.5471. Our algorithm $TONMF$ performed significantly better than all the methods with an AUC of 0.8552. Clearly, this is a significant qualitative difference between the methods. The above three were experiments on real life dataset and we chose market basket for synthetic dataset.

The ROC comparison for the synthetic market basket data is illustrated in Figure 7. In this case, the improvement of the algorithm TONMF over the baseline methods was quite significant. Specifically, the algorithm TONMF had an area under the ROC curve of 0.7598, which is a significant lift. This significantly outperformed the SVD and $RPCA$ method, which had an area under the ROC curve of 0.5731 and 0.5758 respectively. As in the case of the other data sets, the k-NN algorithm performed very poorly with an area under the ROC curve of 0.5431. The consistently poor performance of the k-NN approach over all algorithms is quite striking, and suggests that straightforward generalizations of outlier analysis techniques from other data domains are often not well suited to the text domain.

Based on our conducted experiments on real world and synthetic datasets, we observed that $TONMF$ outperformed every other baseline. Furthermore, the rank of the methods from best to worst is $TONMF, RPCA, SVD$ and NN. Clearly, conventional distance-based methods do not seem to work very well for text data.

4.5 Parameter Sensitivity From (2.3) in Section 2, we can see that the parameters for our algorithm are α,β and the low rank r. We tested the algorithm for different variations in the parameters, and found that our algorithm was insensitive to changes in β. In other words, for a given low rank r and α, the changes in the value of β did not result in significant change in the area under ROC. Hence, in this paper, we provide the charts of the ROC area variation with the parameters α and r on the data sets.

The sensitivity results for the *Reuters* data set are illustrated in Figure 4. The value of α is illustrated on the X-axis, and different values of the low rank r are graphed by different curves in the plot. It is evident in this case, that the area under the ROC increased with increase in low rank r and α. However the improvement started diminishing and changed very marginally at higher ranks r.

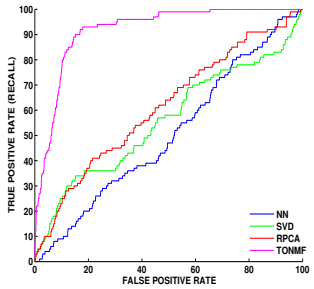

Figure 3: Reuters

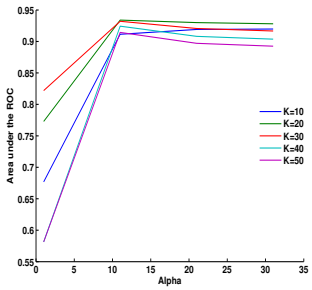

Figure 4: Reuters

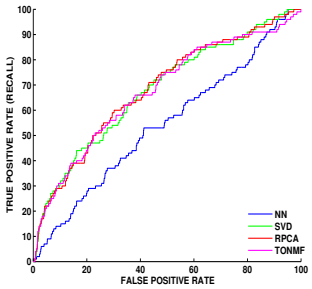

Figure 5: RCV20

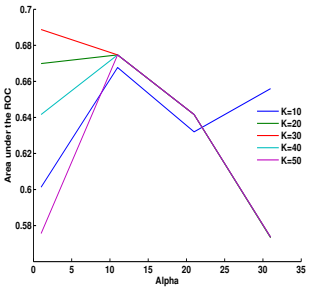

Figure 6: RCV20

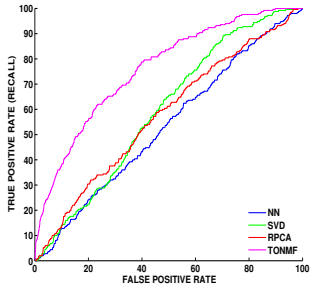

Figure 7: Market Basket

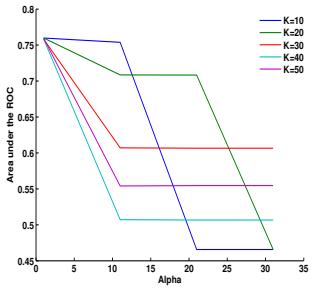

Figure 8: Market Basket

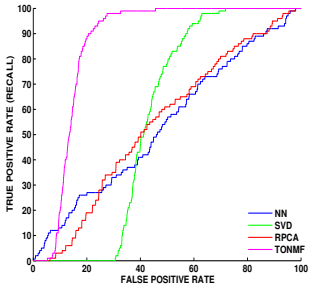

Figure 9: Wiki People

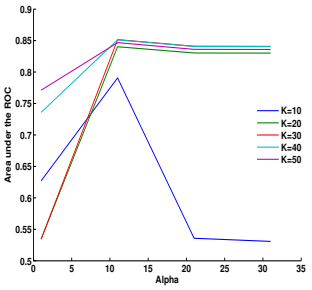

Figure 10: Wiki People

The results for the *RCV20* and *Wiki People* datasets are illustrated in Figure 6 and Figure 10 respectively. As in the previous case, the value of α is illustrated on the X-axis, and different values of the low rank r are represented by different curves. In this case, the area under the ROC curve was relatively insensitive to the parameters. This implies that the algorithm can be used over a wide range of parameters, without affecting the performance too much. Finally, the results for the market basket data set are illustrated in Figure 8. In this case, the area under the ROC curve decreases with increase in low rank r and α. This is because the market-basket data has inherently very low (implicit) dimensionality, and therefore, it is best to use a relatively low rank in order to mine the outliers.

From the parameter sensitivity graphs for real world datasets, we observe that for a given α, the approach is relatively insensitive to the rank of the approximation. It needs to be kept in mind that it is generally faster to determine approximations with lower rank. This implies that, for very large matrices, the algorithm can be made computationally faster by choosing approximations with lower rank without compromising on the performance. According to the model explained in equation (2.3), the parameters α and β balance the importance given to outliers against the matrix sparsity criterion during regularization. By picking $\alpha >> \beta$, the importance of the outlier portion of the regularization increases. From the parameter sensitivity graph, it is evident that for most low ranks K, the increase in the value of α does not improve the performance of the outlier detection. This is because, beyond a particular limit, the weights given to the outlier criterion do not supersede the optimization problem's main objective of extracting the low-rank patterns from the underlying data.

4.6 Further Insights In order to illustrate the inner workings of the matrix factorization approach, we provide some further insights about the statistics buried deep in the algorithm. We also present some interesting observations when outliers share the same vocabulary distribution as regular data points, as is the case for the *RCV20* data set. One observation is that the method of data generation implicitly assumes that all the documents within a "regular" class in a real data set are not outliers. This is of course not true in practice, since some of the documents within these classes will also be outliers, for reasons other than topical affinity. Our algorithm TONMF was also able to detect such distinct documents, much better than the other baseline algorithms. We isolated those false positives of our algorithm TONMF that were not detected in the baselines in the case of the *RCV20* data set. While these outliers officially belonged to one of the regular classes, they did show different *kinds* of distinctive characteristics. For example, while the average number of words in regular documents was 195, the "false positive" outliers chosen by our algorithm were typically either very lengthy with

over 400 words, or were unusually short with less than 150 words. This behaviour was also generally reflected in the number of distinct words per document. Another observation is that these outlier documents typically had a significant vocabulary repetition over a small number of distinct words. Thus, the algorithm was also able to identify those natural outliers, which *ought to* have been considered outliers for reasons of statistical word distribution, as opposed to their topical behaviour.

5 Conclusion

This paper presents a matrix factorization based approach to text outlier analysis. The approach is designed to adjust well to the widely varying structures in different localities of the data, and therefore provides more robust methods than competing models. The approach has the potential to be applied to other domains with similar structure, and as a specific example, we provide experiments on market basket data. We also presented extensive experimental results, which illustrate the superiority of the approach. Our code can be downloaded from `https://github.com/ramkikannan/outliernmf` and tried with any text dataset. We also recommend the readers to check out our extended version [11] of this paper for additional details. In the future, we would like to explore a scalable implementation of our algorithm. The solution is embarrassingly parallelizable, and we would like to experiment it in web scale data. One of the potential extension is incorporating temporal and spatial aspects into the model. Such an extension makes the solution applicable to emerging applications such as topic detection and streaming data.

6 Acknowledgements

This manuscript has been co-authored by UT-Battelle, LLC under Contract No. DE-AC05-00OR22725 with the U.S. Department of Energy. This project was partially funded by the Laboratory Director's Research and Development fund, National Science Foundation (NSF) grant IIS-1348152, Defense Advanced Research Projects Agency (DARPA) XDATA program grant FA8750-12-2-0309 and also sponsored by the Army Research Laboratory (ARL) accomplished under Cooperative Agreement Number W911NF-09-2-0053. Also, H. Woo is supported by NRF-2015R101A1A01061261.

Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the USDOE, DARPA, NSF or ARL.

References

[1] C. Aggarwal. Outlier analysis. *Springer*, 2013.

[2] C. C. Aggarwal and P. S. Yu. Outlier detection for high dimensional data. In *SIGMOD Conference*, pages 37–46, 2001.

[3] S. J. Blasiak, H. Rangwala, and S. Sudarsan. Joint segmentation and clustering in text corpuses. In *SDM*, pages 485–493, 2013.

[4] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander. Lof: Identifying density-based local outliers. In *SIGMOD Conference*, pages 93–104, 2000.

[5] E. J. Candès, X. Li, Y. Ma, and J. Wright. Robust principal component analysis. *JACM*, 58(3):11, 2011.

[6] V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection: A survey. *ACM Comput. Surv.*, 41(3), 2009.

[7] C. Ding, T. Li, and W. Peng. Nmf and plsi: equivalence and a hybrid algorithm. In *SIGIR*, SIGIR '06, pages 641–642, 2006.

[8] C. Ding, D. Zhou, X. He, and H. Zha. r_1-pca: Rotational invariant l_1-norm principal component analysis for robust subspace factorization. In *ICML*, 2006.

[9] E. Esser, X. Zhang, and T. Chan. A general framework for a class of first-order primal-dual algorithm for convex optimization in imaging science. *SIAM J. Imag. Sci.*, 3(4):1015–1046, 2010.

[10] E. Gaussier and C. Goutte. Relation between plsa and nmf and implications. In *SIGIR*, pages 601–602, 2005.

[11] R. Kannan, H. Woo, C. Aggarwal, and H. Park. Outlier detection for text data : An extended version. *CoRR*, abs/1701.01325, 2017.

[12] F. Keller, E. Muller, and K. Bohm. Hics: high contrast subspaces for density-based outlier ranking. In *ICDE*, pages 1037–1048, 2012.

[13] J. Kim, Y. He, and H. Park. Algorithms for nonnegative matrix and tensor factorizations: A unified view based on block coordinate descent framework. *Journal of Global Optimization*, 58(2):285–319, 2014.

[14] E. M. Knorr and R. T. Ng. Algorithms for mining distance-based outliers in large datasets. In *VLDB*, pages 392–403, 1998.

[15] D. Kong, C. Ding, and H. Huang. Robust nonnegative matrix factorization using l21-norm. In *CIKM*, pages 673–682, 2011.

[16] H.-P. Kriegel, P. Kroger, E. Schubert, and A. Zimek. Outlier detection in arbitrarily oriented subspaces. In *ICDM*, pages 379–388, 2012.

[17] A. Lazarevic and V. Kumar. Feature bagging for outlier detection. In *KDD*, pages 157–166, 2005.

[18] D. D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.

[19] J. Liu, S. Ji, and J. Ye. Multi-task feature learning via efficient $l_{2,1}$-norm minimization. In *UAI*, pages 339–348, 2009.

[20] E. Muller, I. Assent, P. Iglesias, Y. Mulle, and K. Bohm. Outlier ranking via subspace analysis in multiple views of the data. In *ICDM*, pages 529–538, 2012.

[21] S. Ramaswamy, R. Rastogi, and K. Shim. Efficient algorithms for mining outliers from large data sets. In *SIGMOD Conference*, pages 427–438, 2000.

[22] A. P. Singh and G. J. Gordon. A unified view of matrix factorization models. In *ECML PKDD*, pages 358–373, 2008.

[23] H. Tong and C.-Y. Lin. Non-negative residual matrix factorization with application to graph anomaly detection. In *SDM*, pages 143–153, 2011.

[24] Y. Wang, J. Yang, W. Yin, and Y. Zhang. A new alternating minimization algorithm for total variation image reconstruction. *SIAM J. Imag. Sci.*, 1(3):248–272, 2008.

[25] H. Xu, C. Caramanis, and S. Sanghavi. Robust pca via outlier pursuit. *IEEE Trans. on Information Theory*, 58(5):3047–3064, 2012.

[26] W. Xu, X. Liu, and Y. Gong. Document clustering based on non-negative matrix factorization. In *SIGIR*, pages 267–273, 2003.

[27] Y. Yang, H. T. Shen, Z. Ma, Z. Huang, and X. Zhou. $l_{2,1}$-norm regularized discriminative feature selection for unsupervised learning. In *IJCAI*, pages 1589–1594, 2011.