ANOMALY DETECTION BASED ON LDA AUTOENCODER AND GAUSSIAN MIXTURE

Frank Xu
Applied Mathematics Dept
Stony Brook University
New York, NY
frank.victo.xu@gmail.com

Xiaoyue Wu
Quantitative Analytics Dept
Credit Suisse
New York, NY
lilywxxah@gmail.com

Yang Xing
Data Science Inst
Columbia University
New York, NY
yx2416@columbia.edu

Gaoyi Shi
Data Science Inst
Columbia University
New York, NY
gs3012@columbia.edu

Xiaoxi Zhen
Data Science Inst
Columbia University
New York, NY
xz2740@columbia.edu

***Abstract*—Anomaly detection is a typical and crucial research area in machine learning and Statistics. It involves many classic methods and state-of-the-art models in supervised learning, unsupervised learning, and semi-supervised learning. This paper presents a novel unsupervised model named LAG based on the Latent Dirichlet Allocation (LDA), Autoencoder (AE) and Gaussian Mixture Model (GMM), which provides a comprehensive solution for the anomaly detection on both structured data and unstructured data. Unsupervised anomaly detection on high-dimensional data is of great importance in both academic research and industrial applications for its outstanding performance and low expense (since no labels are needed for the model training). In this paper, we utilize the LDA to transform the unstructured data into a topic probability vector and use the Autoencoder to generate a low-dimensional representation. Eventually, we leverage the GMM to perform density estimation and anomaly detection. Instead of using decoupled multi-stage training, we jointly optimize the parameters of the AE and GMM simultaneously with a pre-trained LDA. The Experiments indicate LAG outperforms state-of-the-art anomaly detection models with more than 8% F1-score improvement on the public datasets and 5% F1-score improvement on the private dataset.**

***Index Terms*—Anomaly Detection, LDA, Autoencoder, GMM, Joint Optimization, Variational Inference**

I. Introduction

A. Background Introduction

Anomaly detection is a critical problem in many areas, such as information system, medical care, and finance. In terms of the bank industry, anomaly detection or fraud detection is extremely important for Risk management and Compliance. Much outstanding progress has been made in financial anomaly detection in the past few years, most of them are based on the structured data. The development of big data techniques and more advanced analytical tools make it possible to process the unstructured data. However, the unstructured data often involves much higher input dimension, which dramatically increases the difficulty of density estimation. As the core technique of anomaly detection, density estimation identifies the outliers or anomalies by the low-density areas. Therefore, the issue of high-dimension input has to be addressed for the unstructured data based anomaly detection. In this paper, we use the Latent Dirichlet Allocation technique to transform the unstructured data into a low-dimensional vector space, which efficiently addresses the curse of dimensionality. The high dimension issue is not only from the unstructured data but also from structured data. The way to address this issue for structured data includes two primary approaches. First one is a two-step model with dimension reduction in the first step and density estimation in the second step. The disadvantage of this approach is that because the two steps are independently conducted, the dimension reduction step may lose the key information of the anomaly detection step. Therefore, this approach often leads to a suboptimal solution. Another approach is combining the dimension reduction and density estimation by the joint optimization technique which trains the two steps at the same time under a single objective function. This approach can avoid the loss of key information of the original feature space. We use the second approach to process the output of LDA. Aside from the output of LDA we also have other predictive features as the input for Autoencoder. The structure of the entire model is as the following Chart1.

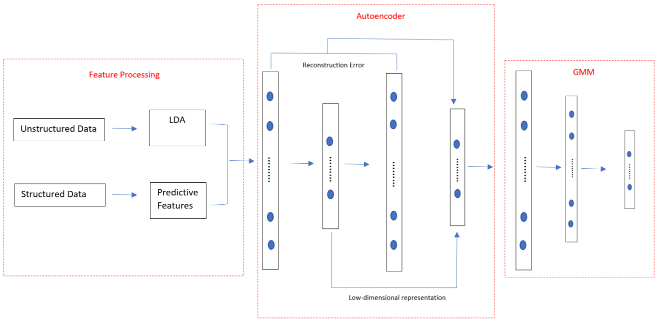

Fig. 1. LAG Structure

B. Related Work

Anomaly detection is broadly used in many industries. Recently, an increasing number of anomaly detection techniques based on the deep neural network was published. The popularity of the deep anomaly detection (DAD) is mainly because the access to large-scale data becomes cheaper and the prediction accuracy of deep neural network models is much higer. Adewumi and Akinyelu [2017] [1] provided a comprehensive survey of deep learning-based methods for fraud detection.

A broad review of deep anomaly detection techniques for cyber-intrusion detection was presented by Kwon et al. [2017] [2]. An comprehensive review of using DAD techniques in the medical care was introduced by Litjens et al. [2017] [3]. Aside from the structured-data based review, many unstructured-data based methods were introduced in the most recent years. The advanced deep learning based methods for video anomaly detection along with various categories have been introduced in Kiran et al. [2018] [4]. The DAD techniques in the applications of speech recognition, video detection, text anomaly detection, etc were presented by Mohammadi et al. [2017] [5] and Ball et al. [2017] [6]. Compared with the above review, one of the contributions of this paper is innovatively utilizing the topic model in NLP to convert the unstructured data which contains the customer behavior information to a dense vector space or low-dimensional space. As the first phase of our deep DAD model, it provides the input data for the subsequent layers. Most machine learning models treat the dimensionality reduction (DR) and clustering independently, but the most current research has shown that the joint optimization of DR and clustering can significantly improve the performance of both. As mentioned in the paper of Bo Yang et al. [2017] [7], many attempts have been conducted to jointly optimize the DR and clustering. In the paper of De Soete & Carroll, [1994] [8]; Patel et al. [2013] [9]; Yang et al. [2017] [10], joint DR and clustering was considered. The rationale is that if there exist some latent space where the entities ideally fall into clusters, then it is natural to seek a DR transformation that reveals such structure, i.e., which yields a low K-means clustering cost. Bo Yang et al. [2017] [7] introduced a joint DR and K-means clustering approach in which DR is realized by learning a deep neural network (DNN).The purpose of this model is to keep the advantage of jointly optimizing the two steps, while exploiting the ability to approximate any nonlinear function of DNN. K-means is a hard clustering technique, but in many cases, the probability of each cluster or segment is critical for anomaly detection. Bo Zong, et al. [2018] [11] proposed a jointly optimized deep autoencoding Gaussian mixture model (DAGMM) for anomaly detection based on the structured data. The model significantly outperforms state-of-the-art anomaly detection techniques, and achieves up to 14% improvement based on the standard F1 score on several public benchmark datasets. The advantage of this model is as follows. Firstly, it avoids the need of pre-training. Secondly, it is more likely to approach the global optima instead of local optima. This paper leverages the structure of Autoencoder and GMM layers from DAGMM and enhances the model by involving the feature processing part with LDA and jointly optimize the parameters of each part, which is an extension of DAGMM on the unstructured data.

II. Methodology

A. Customer Profiling with LDA

An increasing requirement of the anomaly detection and fraud detection in the financial industry makes it urgent to broaden the scope of data and explore more techniques to meet the requirement. From the bank perspective, the traditional customer profiling data is not sufficient for the real-time anomaly detection since the update frequency of such data is not in real-time. The ideal data is from the raw message of each transaction. However, such data is a typical text data and is very unstructured. How to utilize this data to construct a more predictive model becomes a critical issue. In this paper, we leverage the NLP topic model Latent Dirichlet Allocation (LDA) to convert the unstructured data to a topic probability vector. We also attempt to perform other topic models and dimension reduction models in this phase as the benchmark models such as Non-negative matrix factorization (NMF) and PCA. It turns out that using LDA as the dimension reduction method in the entire LAG model has a better performance than the other methods based on the experiments with real-world data.

1) Construct Text Corpora: The LDA model is designed for the NLP problem. It requires three basic components, words, documents and corpora. The first challenge is how to convert the financial transactions to such components. This paper provides a simple way to perform the conversion. A single transaction is a raw message text. The message text can be tokenized as a list of tokens according to which sort of information should be collected. For instance, a raw message text contains the transaction type, payment method, transaction amount and transaction date, etc in which we can only tokenize the transaction amount and the transaction type since these are predictive features for anomaly detection. By concatenating all generated tokens as a single string, we can convert a transaction to a word. Notice that all the tokens should be a categorical value. For the continuous value such as transaction amount should be bucketed to groups. A document consists of a batch of transactions of a specific customer, which represents the financial behavior of this customer. The corpora consists of all documents eventually.

2) Efficiency of LDA: LDA is a generative probabilistic model for the text or discrete data designed by Andrew Y. Ng et al. [2003] [12]. LDA is also a typical dimension reduction technique, which can convert a document to a topic probability vector. Given a set of documents in a corpus, each of which contains a set of words, LDA discovers K topics for each document, represented as latent factors each taking the form of a distribution of words. The LDA model assumes a generative process for each document d that its topics Z_{dn} follow Multinomial(θ_d) drawn from Dirichlet $\theta_d \sim$ Dir(α), and each word w_{dn} is chosen from selected topic by $w_{dn} \sim$ Multinomial (β_{zn}) drawn from Dirichlet $\beta_k \sim$ Dir(η). Two major algorithms for inferring the model parameters are variational inference and gibbs sampling. The former is a simplified approximation method that results in a biased estimate, but is computationally efficient with well-defined numerical convergence criteria, while the latter is a probabilistic technique that converges to an unbiased solution, but its convergence is hard to diagnose. Here we discuss briefly the variational inference EM algorithm, which is used in our LAG model. In the variational inference EM algorithm, the intractable distribution of the hidden variables given the data $P(\theta, z|w, \alpha, \beta)$ is approximated by a simplified, conditionally independent variational distribution as shown below:

$$Q(\theta, z|\gamma, \phi) = \prod \int_{d=1}^{M} [q(\theta_d|\gamma_d) \prod \int_{n=1}^{N_d} [P(z_{dn}|\phi_{dn})]] \quad (1)$$

Where γ and $\{\phi_1, \cdots, \phi_{Nd}\}$ are free variational parameters. In the E-step, given fixed parameter inputs α and β, optimal values of γ_d and ϕ_d are found using the algorithm as follows.

Algorithm 1 E-step with fixed α and β inputs

```
initialize $\phi^0_{ni} := 1/K$ for all word n and topic $i$
initialize $\phi_i := \alpha_i + N/K$ for all topic $i$
repeat
for $n = 1$ to $N_d$ do
    for $i = 1$ to $K$ do
        $\phi^{t+1}_{ni} := \beta_{iw_n} exp(\Psi(\phi^t_i))$
    end for
    normalize $\phi^{t+1}_n$ to sum to 1
end for
$\gamma^{t+1} := \alpha + \sum^N_{n=1} \phi^{t+1}_n$.
until converge
```

Where Ψ is the Digamma function. ϕ and γ are the variational parameters as defined previously.The algorithm indicates a requirement of $O(N_d K)$ operations for a single document. Since empirical experiment suggests the number of iterations required is in the order of N_d to achieve convergence, the total number of operations is roughly $MN^2_d K$ for a corpus of size M.

The M-step is completed by maximizing the resulting lower bound on the log likelihood with respect to α and β. The two steps repeat until the lower bound of likelihood $\ell(\alpha, \beta) = \sum^M_{d=1} [logp(w_d|\alpha, \beta)]$ converges[1]. The complexity of the M-step is $O(VK)$, where V is the total vocabulary in the corpus.

When applied to customer transaction data, we regard a batch transactions of each customer as a document and each transaction as a word, LDA then learns the topics (patterns) of the customer behaviors, formed by the type of transactions they make. Notice that V is the vocabulary size, which is also the number of feature dimensions of the TF-IDF Matrix. As the number of unique words[2] of a single document N_d, we can conclude $N_d < V$. Therefore, the time complexity of the corpus $MN^2_d K < MV^2 K$. Based on the following research of Dave Blei[3], the complexity of LDA by the mean-field variational inference is actually MVK instead of MV^2K. In the following section, we will use m as the number of feature dimensions of the TF-IDF Matrix of text corpus, n as the number of data or documents and k as the number of topics. Thus, the time complexity of LDA is nmk.

TABLE I
TIME COMPLEXITY COMPARISON FOR DIFFERENT DR METHODS

DR Methods	Time Complexity
LDA	$O(nmk)$
NMF	$O((nm)^{2^t t^2})$
PCA	$O(m^3)$
Kernal-PCA	$O(m^3)$

3) Advantages of LDA: The traditional bag of words method will convert all unique words of the corpora as features and represents a document in a high-dimensional feature space. The high dimension issue will bring troubles for anomaly detection. When the dimensionality of input data becomes higher, it is more difficult to perform density estimation in the original feature space, as any input sample could be a rare event with a low probability to observe as referred by the research of Chandola et al. [2009] [13]. Conventional dimension reduction methods for anomaly detection include SVD, PCA and kernel PCA with implicit non-linear projections induced by specific kernels. The time complexity of PCA and kernel PCA is $O(m^3)$, here m is the number of feature dimension of the TF-IDF Matrix of our transaction text corpora. Compared with the time complexity of LDA from the Table1, PCA is less computationally efficient than LDA. Also, some research indicates that PCA based anomaly detection methods are sensitive to the noise. A significant amount of anomalous samples could also lurk with a normal level of error by using PCA if the data are noisy.

Non-negative matrix factorization (NMF) is another popular topic model, which is based on the matrix decomposition and optimization techniques. As an NP-hard problem, the time complexity of NMF is thus dependent on the stopping criteria of iterative updates. Suppose t iterations are implemented, m is the number of feature dimension of TF-IDF Matrix and n is the number of data or documents. The NMF is subject to an $(nm)^{O(2^t t^2)}$ complexity at a worst-case scenario. At an optimal case, the NMF requires $(2^t nm)^{O(t^2)}$. Because of its strong capability in information compression, NMF has been widely applied in applications such as facial and voice recognition, and text mining problems. However, compared with the time complexity of LDA, NMF is much less efficient than LDA.

The dimension reduction as the first and the most important step in anomaly detection is very easy to lose the key information for detecting the anomalies from normal samples as referred by the research of Bo Zong et al. [2018] [11]. Such an issue is caused by two difficulties: First is independent optimization for dimension reduction and density estimation. Second is the inappropriate dimension reduction method. The first one can be addressed by joint optimization, which will be explained thoroughly in the following section. The second difficulty can be only addressed by attempting different approaches and comparing the performance of models

[1] Again variational inference provides a tractable lower bound on the log likelihood, which can be maximized with respect to α and β.

[2] In the original LDA paper, N_d represents the number of all words in a single document. However, it turns out that N_d can be the number of unique words instead based on the following research of LDA.

[3] The reason is that, in a document, we need not compute posterior multinomials for each instance of each word but only once for each unique word in that document. In the LDA paper, editors did not write things down that way to make the math simpler.

with different dimension reduction methods. Based on the experiments conducted in the following section, it turns out that LDA as the dimension reduction method in the entire model has the best performance with more than 6% improvement at F1-score compared with other dimension reduction methods.

B. The Equivalence of GMM and LMM

The Log-linear model is a discriminative model estimating posterior probabilities of class $\boldsymbol{c}$ given the feature vector $\boldsymbol{x} \in \mathbb{R}^{\mathrm{M}}$:

$$p_\theta(c|x) = \frac{\exp\left(w_c^T f(x) + b_c\right)}{\sum_{c'} \exp\left(w_{c'}^T f(x) + b_{c'}\right)} \tag{2}$$

where the parameter set $\theta = \{w_c, b_c\}$, and the specific parameters $w_c \in \mathbb{R}^K$, $b_c \in \mathbb{R}$. The feature function $f(x) : \mathbb{R}^M \to \mathbb{R}^N$ corresponds to the mapping function such as linear, polynomial or any non-linear feature mapping.

The multivariate Gaussian density function of class $\boldsymbol{c}$ given the feature vector $\boldsymbol{x} \in \mathbb{R}^{\mathrm{M}}$:

$$\mathcal{N}\left(x|\mu_c, \Sigma_c\right) = \frac{1}{|2\pi\Sigma_c|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}\left(x_{\mu_c}\right)^\top \Sigma_c^{-1}\left(x_{\mu_c}\right)\right) \tag{3}$$

where the parameter set is$\{\mu_c, \Sigma_c\}$, μ_c is the mean vector, Σ_c is the covariance matrix, and $x_{\mu_c} = x - \mu_c$. The joint probability of the single Gaussian model includes the class prior $p(c) \in \mathbb{R}$ is defined as: $p(x, c) = p(c)\mathcal{N}\left(x|\mu_c, \Sigma_c\right)$. Using the Bayes rule the posterior can be derived as $p(c|x) = \frac{p(x,c)}{\sum_{c'} p(x,c')}$. Therefore, the class posteriors induced by the single Gaussian model is:

$$p_{\text{Gauss}}(c|x) = \frac{p(c)\mathcal{N}\left(x|\mu_c, \Sigma_c\right)}{\sum_t p\left(c'\right)\mathcal{N}\left(x|\mu_c, \Sigma_c\right)} \tag{4}$$

By the following transformation as showed in Table2, we can convert the Gaussian posterior to a Log-linear posterior.

$$p_{\log}(c|x) = \frac{\exp\left(x^\top \Lambda_c x + \lambda_c^\top x + \alpha_c\right)}{\sum_{c'} \exp\left(x^\top \Lambda_c x + \lambda_{c'}^\top x + \alpha_{c'}\right)} \tag{5}$$

Therefore, we have the first lemma as follows:

$\boldsymbol{Lemma1}$: The posterior log-linear model in the equation 5 and posterior Gaussian model in the equation 4 are equivalent.

TABLE II
PARAMETERS TRANSFORMATION FROM GAUSSIAN INTO LOG-LINEAR

Gaussian	Log-linear
Λ_c	$-\frac{1}{2}\Sigma_c^{-1}$
λ_c	$\Sigma_c^{-1}\mu_c$
α_c	$-\frac{1}{2}\left(\mu_c^\top\Sigma_c^{-1}\mu_c + \log\|2\pi\Sigma_c\|\right) + \log p(c)$

The Gaussian mixture model (GMM) is defined as the superposition of Gaussian densities with the mixture weights $p(l|c) \in \mathbb{R}, 0 < p(l|c) < 1, \Sigma_l p(l|c) = 1$. For all c, we have the joint distribution as $p(x, c) = \sum_l p(l|c)\mathcal{N}\left(x|\mu_{cl}, \Sigma_{cl}\right)$. Similar to the single Gaussian posterior, the class posterior of GMM is as follows:

$$p_{\text{GMM}}(c|x) = \frac{p(c)\sum_l p(l|c)\mathcal{N}\left(x|\mu_{cl}, \Sigma_{cl}\right)}{\sum_{c'} p\left(c'\right)\sum_l p\left(l|c'\right)\mathcal{N}\left(x|\mu_{c'l}, \Sigma_{c'l}\right)} \tag{6}$$

By the transformation in Table3, the above GMM posterior can be represented as a log-linear model:

$$p_{LMM}(c|x) = \frac{\sum_l \exp\left(x^\top \Lambda_{cl} x + \lambda_{cl}^\top x + \alpha_{cl}\right)}{\sum_{c'} \exp\left(x^\top \Lambda_{cl} x + \lambda_{el}^\top x + \alpha_{c'l}\right)} \tag{7}$$

Such a log-linear model shall be referred to as a log-linear mixture model (LMM). We have the second lemma as follows:

$\boldsymbol{Lemma2}$: The posterior LMM in the equation 7 and posterior GMM in the equation 6 are equivalent.

TABLE III
PARAMETERS TRANSFORMATION FROM GMM INTO LMM

GMM	LMM
Λ_{cl}	$-\frac{1}{2}\Sigma_{cl}^{-1}$
λ_{cl}	$\Sigma_{cl}^{-1}\mu_{cl}$
α_{cl}	$-\frac{1}{2}\left(\mu_{cl}^\top\Sigma_{cl}^{-1}\mu_{cl} + \log\|2\pi\Sigma_{cl}\|\right) + \log p(c)$

The Softmax layer uses a Softmax function to normalize the input data to a vector that each component will be in the interval $(0, 1)$, so that they can be interpreted as probabilities. Assume the input vector $\mathbf{z} = (z_1, \ldots, z_K) \in \mathbb{R}^K$ has K dimension, then each component of vector $\mathbf{z}$ after transformed by Softmax function is as follows:

$$\mathbf{Softmax}(\mathbf{z})_i = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}} \tag{8}$$

Compare the above Softmax function with the Log-linear or LMM posterior, we can easily understand that they are equivalent. That's why we can use a Softmax layer to represent a GMM posterior. Therefore, we have following theorem:

$\boldsymbol{Theorem}$: The Log-linear model in equations 2,5,7 corresponds to the Softmax output layer. The network from the Autoencoder up to the output of the last hidden layer of the entire model forms the feature function f in the equation 2.

C. Joint Optimization for Autoencoder & GMM

As mentioned in the section Advantage of LDA, these approaches that separate dimension reduction step and density estimation step will lead to a suboptimal solution because dimensionality reduction is unaware of the subsequent density estimation and the key information for anomaly detection can be removed in the first place, Bo Zong et al. [2018] [11]. Therefore, it is desirable to combine two steps together and jointly optimize the parameters of both steps. In this section, we propose a deep Autoencoder combined with a multi-layer network of GMM, whcih is mainly based on the DAGMM from the research of Bo Zong. Our LAG model can be seen as an extension of DAGMM for unstructured data. However, the advantage of LAG over DAGMM is not only the capability of handling unstructured data but also joint optimization on LDA and DAGMM, which will be briefly explained in the section of future work. In this section, we only focus on the joint optimization of Autoencoder and GMM.

The Autoencoder and LDA preserve the key information of customer behavior pattern in a low-dimensional space.

The function of such Autoencoder includes extracting key information for anomaly detection and providing reconstruction error for each sample. Assume X is the input of Autoencoder, y_l is the low-dimensional representations, $y_l = f_e(X|w_e, b_e)$, w, b are the parameters of encoder, f_e is the encoder function. In the decoder step, Autoencoder can generate an output $X^{'}$ to represent the input X. $X^{'} = f_d(y_l|w_d, b_d)$, w_d, b_d are the parameters of decoder, f_d is the decoder function. The reconstruction error is the L_2 distance between X and $X^{'}$, which is denoted by $y_r = \|\mathbf{X} - \mathbf{X}'\|_2^2$. Therefore, the output of Autoencoder or the input of GMM is as follows:

$$y = [y_l, y_r] \tag{9}$$

$$y_l = f_e(X|w_e, b_e), y_r = \|\mathbf{X} - \mathbf{X}'\|^2 \tag{10}$$

Given the low-dimensional representations y and integer K as the number of mixture components, GMM can perform the density estimation. Assume the GMM mixture-component distribution is θ, mixture means μ, and mixture covariance matrix Σ. Instead of using EM to estimate the θ, we use a multi-layer network with the last Softmax layer to perform the estimation. The foundation of this method has been explained in the previous section. the estimation of GMM is as follows:

$$z = f_{MLN}(y|w_g, b_g) \tag{11}$$

$$\alpha = \mathbf{Softmax}(z) \tag{12}$$

Where α represents the predicted soft mixture-component membership of GMM. w_g, b_g is the parameters of the multi-layer network. f_{MLN} is the multi-layer network function. Given a batch of N samples and the mixture-component membership prediction $\hat{\alpha}$, we can estimate the parameters of the kth component of GMM as follows:

$$\hat{\theta}_k = \sum_{i=1}^{N} \frac{\hat{\alpha}_{ik}}{N} \tag{13}$$

$$\hat{\mu}_k = \frac{\sum_{i=1}^{N} \hat{\alpha}_{ik} \mathbf{y}_i}{\sum_{i=1}^{N} \hat{\alpha}_{ik}} \tag{14}$$

$$\hat{\Sigma}_k = \frac{\sum_{i=1}^{N} \hat{\alpha}_{ik} \left(\mathbf{y}_i - \hat{\mu}_k\right)\left(\mathbf{y}_i - \hat{\mu}_k\right)^T}{\sum_{i=1}^{N} \hat{\alpha}_{ik}} \tag{15}$$

Notice that $\widehat{\alpha_{ik}}$ is the membership estimation for the kth component based on the ith sample. The $\widehat{\mu_k}, \widehat{\Sigma_k}$ is the estimated paramaters for the kth Gaussian distribution. The likelihood function of GMM can be further represented as follows:

$$\mathbf{L} = \sum_{k=1}^{K} \hat{\theta}_k \frac{\exp\left(-\frac{1}{2}\left(\mathbf{y} - \hat{\mu}_k\right)^T \hat{\mathbf{\Sigma}}_k^{-1} \left(\mathbf{y} - \hat{\mu}_k\right)\right)}{\sqrt{\left(2\pi\right)^N \left|\hat{\mathbf{\Sigma}}_k\right|}} \tag{16}$$

The anomaly evaluation is usually based on the likelihood function or energy function. Specifically, in the testing phase, when the likelihood of a testing example is very small or the energy value of this sample is very big, we can conclude that the testing sample is likely to be an anomaly. Given a pre-determined threshold, all samples can be easily classified as normal or anomaly. Conventionally, the energy function is used more often, which is expressed as $\mathbf{E} = -log\left(\mathbf{L}\right)$.

Instead of performing optimization independently for Autoencoder and GMM, we jointly optimize all parameters by mixing the objective functions together. The parameter set is $\mathbf{p} = (w_e, b_e, w_d, b_d, w_g, b_g, \mu_k, \Sigma_k, \theta_k)$, $k = 1..K$. The joint objective function $\mathbf{J_P}$ is as follows:

$$\frac{1}{N}\sum_{i=1}^{N} \left\|\mathbf{X_i} - \mathbf{X_i}'\right\|^2 + \frac{\lambda_1}{N}\sum_{i=1}^{N} E_i + \lambda_2 \sum_{i=1}^{N}\sum_{j=1}^{d} D(\Sigma_{jk}) \tag{17}$$

Notice that the E_i is the energy value of the ith sample. λ_1, λ_2 are the hyperparameters for penalty items. GMM usually has the singularity problem, which is caused by the trivial solutions. The trivial solution are triggered when the diagonal entries in the covariance matrices degenerate to 0. To avoid this problem, we add an extra penalty item which is the sum of diagonal entries of all covariance matrix, where $D(\Sigma_{jk})$ is the sum of diagonal entries for matrix Σ_{jk}. This penalty item can avoid the trivial solution since small diagonal entries will cause a big value of this item.

III. Experiments

In this section, we select some most popular public benchmark datasets to evaluate the effectiveness of our LAG model and compare the model performance of LAG with some state-of-the-art unsupervised anomaly detection models. Also, we will introduce the detailed training strategy, optimization tricks, model configuration, etc of this model.

A. Public DataSet

We select three benchmark datasets: 20 Newsgroups, AG News and Reuters-21578. The rationale of such selection is the selected datasets are text datasets with many experiment results. Also, as the datasets for text or topics classification, they are very suitable for our model. The description of the datasets is as follows:

- **20 Newsgroups**: The 20 Newsgroups dataset is a collection of approximately 20,000 newsgroup documents. The data is organized into 20 different newsgroups, each corresponding to a different topic.
- **AG News**: The AG's news topic classification dataset is constructed by choosing 4 largest classes from the original corpus. Each class contains 30,000 training samples and 1,900 testing samples. The total number of training samples is 120,000 and testing 7,600.
- **Reuters-21578**: The Reuters-21578 dataset contains a collection of documents that appeared on Reuters newswire in 1987. The documents were assembled and indexed with categories. This dataset is appropriate for testing natural language processing and information retrieval algorithms. It has 8000 documents and 15,818 words.

For anomaly detection instead of text classification, we need to perform some adjustments on the datasets. Firstly, we need to separate the text documents with different topics or categories into normal groups and anomaly groups. The strategy is that hold all documents under a specific topic or category as the normal documents and randomly select documents from the rest of the topics or categories as the anomaly documents. Consider the topic is hierarchical, we choose the low-level topics to generate samples since the low-level topics contain more documents for training the neural networks. Another strategy to expand data is dividing each document into several pieces, each piece shares the same topic. In our experiment, we divide each document into four pieces. The tweaked dataset is displayed in Table4[4].

TABLE IV
STATISTICS OF ADJUSTED 20 NEWSGROUPS

Normal Topic	#Document	#Normal	#Anomaly	Ratio
hardware	5312	21248	1338	0.063
autos	1986	7944	381	0.048
sports	1993	7972	263	0.033
science	5233	20932	1004	0.048
religion	1625	6500	403	0.062
politics	3492	13968	712	0.051

All the public datasets are adjusted in the way as described above. Notice that because of the extreme imbalance of the real-world data of anomaly detection, we need to keep the anomaly ratio very low to meet the requirement, which is shown in the last column of Table4. In all cases, we hold out 10% of the data for testing purpose and trained the models on the rest of 90% data. The final metrics for evaluating the model performance will be averaged by the metric on each topic or category. For instance, the final F1-score on the 20 Newsgroups dataset will be the average value of the F1-score of each topic in the Table4. Moreover, the stop words will be removed before training the model and the optimization method we used is SGD.

B. Private Dataset

The purpose of this paper is to provide an anomaly detection solution for the transaction text data. Because of the confidentiality of financial transactions, there is few public data for academic research. Therefore, we exploit the private dataset from a global bank to validate the capability of suspicious transaction detection and suspicious customer behavior detection of our model. We only post the model performance results in Table5. Any sensitive information about companies and customers will not appear in this paper. The private data consists of 30 million raw transaction messages which contain 100 thousand confirmed suspicious transactions and 50 thousand unqiue customers. Each transaction contains many key-information such as amount value, transaction type, payment method, originator's name, and beneficiary's name, etc. Similar to the public dataset, we keep 10% data for testing purpose and the rest 90% for training the model.

TABLE V
MODEL COMPARISON OF PRIVATE DATASET

	Transactions		
	Precision	Recall	F1-score
LAG	0.9484	0.7502	0.8357
PCA+DGMM	0.8866	0.6997	0.7734
Kernel-PCA+DGMM	0.8637	0.6746	0.7386
NMF+DGMM	0.8743	0.7307	0.7855
LDA+OC-SVM	0.8638	0.6155	0.7085
PCA+OC-SVM	0.8825	0.6812	0.7561
Kernel-PCA+OC-SVM	0.8770	0.7127	0.7806
NMF+OC-SVM	0.8712	0.5498	0.6505

C. Model Comparison

We select some of the state-of-the-art models as the benchmarks including NMF, PCA, kernel-PCA as the alternatives of LDA in the dimension reduction step and DGMM, OC-SVM in the density estimation step. Notice that the alternatives of LDA need to be performed on the TF-IDF Matrix, only LDA can be directly performed on the raw data. For all public datasets, the anomalies are taken as positive samples. For the private dataset, suspicious customers are taken as positive samples. The metrics we have used are Precision, Recall, and F1-score. The predicted anomaly samples in the testing phase are determined by the sample energy as described in the Methodology section. The sample with top-ranked energy will be detected as an anomaly. Different thresholds will generate different confusion matrices and F1-scores. We choose the best threshold based on the highest F1-score. The following Table5 shows the averaged metrics values for the private dataset, where the metrics for LAG outperforms the other models. It achieves the 5% improvement at F1-score compared with the best model. Table6 shows 8% improvement at F1-score on the public datasets. Only in the AG News experiment LAG doesn't achieve the highest F1-score, but the score is still good.

IV. Future Work

The ideal solution for our project is an end-to-end fashion of LAG, where we could perform joint optimization on LDA, Autoencoder, and GMM. However, the current LAG model is only achieved by joint optimization on Autoencoder and GMM. In the current phase, pre-training is necessary for LDA. The difficulty for the ideal solution is how to put the deep neural network and LDA in the same framework and consider the entire optimization problem from the perspective of Variational Inference.

V. Conclusion

In this paper, we propose a novel unsupervised anomaly detection model named LAG structured by LDA, Autoencoder, and GMM. The innovation of this paper includes the following aspects: Firstly, we propose a way to conduct tokenization for the transaction data, which can convert a transaction to a word and a batch of transactions to a document that represents the financial behavior of a customer. Secondly, we provide a way to deal with the unstructured data by exploiting LDA,

[4] We only choose the dominant topics with sufficient documents and unique words to meet the requirement of deep neural network training.

TABLE VI
MODEL COMPARISON OF PUBLIC DATASET

	20 Newsgroups			AG News			Reuters-21578		
	Precision	Recall	F1-score	Precision	Recall	F1-score	Precision	Recall	F1-score
LAG	0.9710	0.7574	0.8270	0.7940	0.5632	0.6588	0.8245	0.6803	0.7410
PCA+DGMM	0.8893	0.5728	0.6879	0.7345	0.4802	0.5637	0.7981	0.5032	0.5714
Kernel-PCA+DGMM	0.9015	0.6494	0.7412	0.7757	0.5839	0.6544	0.8065	0.5796	0.6652
NMF+DGMM	0.8593	0.6259	0.7162	0.7918	0.5831	0.6556	0.7911	0.5942	0.6563
LDA+OC-SVM	0.8841	0.5888	0.7013	0.7883	0.5867	0.6581	0.7163	0.4273	0.4848
PCA+OC-SVM	0.8360	0.5975	0.6795	0.8044	0.6194	0.6857	0.7957	0.5114	0.6033
Kernel-PCA+OC-SVM	0.8508	0.5436	0.6546	0.8097	0.5379	0.6151	0.7666	0.4924	0.5848
NMF+OC-SVM	0.8981	0.5692	0.6883	0.8186	0.5742	0.6652	0.7902	0.5873	0.6606

which can transform text data or any discrete data into a low-dimensional space and the low-dimensional topic vector generated by LDA will be very helpful in the downstream tasks. Thirdly, we combine the LDA, Autoencoder, and GMM as an entire model to perform anomaly detection. This novel model shows outstanding performance on many benchmark datasets. The reason for the outstanding performance based on our analysis is due to the advantage of LDA in dimension reduction and the advantage of joint optimization for the followed neural networks, where we can keep the key information for anomaly detection.

REFERENCES

[1] Aderemi O Adewumi and Andronicus A Akinyelu. A survey of machine-learning and nature-inspired based credit card fraud detection techniques. International Journal of System Assurance Engineering and Management, 8(2): 937–953, 2017.

[2] Donghwoon Kwon, Hyunjoo Kim, Jinoh Kim, Sang C Suh, Ikkyun Kim, and Kuinam J Kim. A survey of deep learning-based network anomaly detection. Cluster Computing, pages 1–13, 2017.

[3] Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen Awm Van Der Laak, Bram Van Ginneken, and Clara I S´anchez. A survey on deep learning in medical image analysis. Medical image analysis, 42:60–88, 2017.

[4] B Ravi Kiran, Dilip Mathew Thomas, and Ranjith Parakkal. An overview of deep learning based methods for unsupervised and semi-supervised anomaly detection in videos. arXiv preprint arXiv:1801.03149, 2018.

[5] Shahriar Mohammadi and Amin Namadchian. A new deep learning approach for anomaly base ids using memetic classifier. International Journal of Computers, Communications Control, 12(5), 2017.

[6] John E Ball, Derek T Anderson, and Chee Seng Chan. Comprehensive survey of deep learning in remote sensing: theories, tools, and challenges for the community. Journal of Applied Remote Sensing, 11(4):042609, 2017.

[7] Bo Yang, Xiao Fu, Nicholas D Sidiropoulos, and Mingyi Hong. Towards k-means-friendly spaces: Simultaneous deep learning and clustering. In International Conference on Machine Learning, 2017a.

[8] De Soete, G. and Carroll, J. D. K-means clustering in a low-dimensionaleuclideanspace. InNewApproachesin ClassificationandData-Analysis,pp.212–219.Springer, 1994.

[9] Patel, V. M., Van Nguyen, H., and Vidal, R. Latent space sparse subspace clustering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 225–232, 2013.

[10] Yang, B., Fu, X., and Sidiropoulos, N. D. Learning from hidden traits: Joint factor analysis and latent clustering. IEEE Transaction on Signal Processing, pp. 256–269, Jan. 2017.

[11] Bo Zong, Qi Song, Martin Renqiang Min, Wei Cheng, Cristian Lumezanu, Dae-ki Cho, Haifeng Chen: Deep Autoencoding Gaussian Mixture Model for Unsupervised Anomaly Detection. ICLR (Poster) 2018

[12] David Blei, Andrew Ng, and Michael Jordan. 2003b. Latent Dirichlet Allocation. Journal of Machine Learning Research, 2003(3):993–1022.

[13] Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. ACM Comput. Surv, 41:15:1–15:58, 2009.