IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 27, 2018, accepted May 31, 2018, date of publication June 18, 2018, date of current version July 12, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2848298

Sentiment Classification Based on Information Geometry and Deep Belief Networks

MENG WANG, ZHEN-HU NING, CHUANGBAI XIAO, AND TONG LI
Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China

Corresponding author: Zhen-Hu Ning (ningzhenhu@126.com)

This work was supported in part by the Beijing Science and Technology Planning Program of China under Grant Z171100004717001, in part by the Beijing Natural Science Foundation under Grant 4162007, in part by the Natural Science Foundation of China under Grant 61501008, in part by the National Natural Science Foundation of China under Grant 61501007, in part by the Beijing Postdoctoral Research Foundation under Grant 2017-22-030, and in part by the Beijing Science CCF-Venustech Open Research Fund under Grant CCF-VenustechRP2017008.

ABSTRACT Sentiment classification for reviews has attracted increasingly more attention from the natural language processing community. By embedding prior knowledge into learning structures, classifiers often achieve a better performance than original methods. In this paper, we propose a sophisticated algorithm based on deep learning and information geometry in which the distribution of all training samples in the space is treated as prior knowledge and is encoded by deep belief networks (DBNs). From the view of information geometry, we construct the geodesic distance between the distributions over the features for classification. The study of the distributions contributes to the training of the DBN, since the distance is correlated to the error rate in the classification. Finally, we evaluate our proposal using empirical data sets that are dedicated for sentiment classification. The results show that our algorithm results in a significant improvement over existing methods.

INDEX TERMS Information geometry, neural networks, semi-supervised learning, sentiment classification.

I. INTRODUCTION

Due to the fast development of the internet, people are able to express their opinions much easier and in different ways. As a result, it is not surprising that currently there are tons of reviews available. Sentiment classification for such reviews has attracted increasingly more attention from the Natural Language Processing (NLP) community.

Sentiment classification refers to the use of natural language processing, text analysis and computational linguistics to identify and extract subjective information from source materials. Sentiment classification aims to determine the attitude of a speaker with respect to some topic or the overall contextual polarity of a document, such as 'positive' or 'negative' and 'thumbs up' or 'thumbs down' [1]. Methods for document sentiment classification are generally based on the lexicon and corpus [2]. The lexicon-based approaches can derive a sentiment measure for text based on sentiment lexicons. The corpus-based approaches involve a statistical classification method. The corpus-based approaches usually outperform the lexicon-based approaches and have been used in unsupervised learning, supervised learning and semi-supervised learning.

Early research within this field includes the works of Pang *et al.* [3] and Turney [4]. They applied supervised learning and unsupervised learning for classifying the sentiments of movie reviews and automobile reviews respectively. The study of supervised learning methods for sentiment classification began with the work in [3]. On the basis of Markov Logic Networks (MLNs), a study proposed a cross-domain multitask text sentiment classification method rooted in transfer learning [5]. These methods are widely used in analyzing the sentiments of various topics, such as movie reviews [6], micro-blogs [7], [8] and so on. The idea is to train a domain-specific sentiment classifier for each target domain using the labeled data in that domain. Furthermore, some scholars apply machine learning approaches to derive a classifier through supervised learning [9], [10]. In 2018, a platform was presented to automate the processing of information obtained from social networks by focusing on improving the accuracy of decision support systems for sentiment analysis [11]. In addition, a feature selection method was introduced to improve the performance of the supervised learning algorithms [12]. Although these methods have good performance, they all rely on labeled data as the training set,

which is normally difficult to obtain. Even though several works use the domain adaptation approach [13]–[16], as is the challenge of the domain-specific approach, annotating a large scale corpus for each domain is very expensive. Unsupervised learning for sentiment classification maximizes the likelihood of observed data without any labeled reviews [17]. In [4], the classification of a review is predicted by the average semantic orientation of the phrases in the review that contain adjectives or adverbs. In addition, a phrase has a positive semantic orientation when it has good associations (e.g., ''subtle nuances'') and a negative semantic orientation when it has bad associations (e.g., ''very cavalier''). Semantic Orientation from Pointwise Mutual Information (SO-PMI) [18] is a method for inferring the semantic orientation of a word from its statistical association with a set of positive and negative paradigm words. Read and Carroll investigated the effectiveness of word similarity techniques when performing weakly supervised sentiment classification [19]. In 2014, a novel learning model based on active learning and self-training was introduced to incorporate unlabeled data from the target language into the learning process for cross-lingual sentiment classification [20]. Because labeled data are not used by unsupervised learning approaches, they are expected to be less accurate than those based on supervised learning [21]. Several semi-supervised learning approaches have been proposed, which use a large amount of unlabeled data together with labeled data for learning [22], [23]. Zhou *et al.* [24] proposed a semi-supervised learning algorithm called fuzzy deep belief networks for sentiment classification, which is based on the deep learning algorithm Deep Belief Networks (DBNs) [25] and fuzzy sets [26]. However, it does not learn any information from the labels during its unsupervised learning (the learning of RBMs).

Information geometry has found various applications in many fields, such as the asymptotic theory of statistical inference [27], semiparametric statistical inference [28], and the Expectation–Maximization (EM) algorithm [29].

In this paper, we propose to enhance DBNs with information geometry in order to address the aforementioned challenges. Our method guides the learning of RBMs to absorb information from the labels. The details are described as follows. The neural networks discriminate the pattern through learning its distribution. We treat the value of a feature as a variable. We describe its value by its distribution. In our case, the patterns are divided into two categories. In each category, one feature is associated with a distribution. Then, it has two distributions for two categories. The distributions are calculated based on labeled data, which makes use of the information from the labels. With the theory of information geometry, the distributions of the features of all labeled training samples in the space are mapped into a statistical manifold. On the manifold, the geodesic distance is an effective measurement of the difference among the distributions. Although there are many other alternative measures [30], the measurement provided by information geometry has many good properties, such as information monotonicity [31] and being invariant under coordinate transformations [32]. If the distance between the distributions of a feature is larger, then the feature is more discriminative (important), not the other way around. This is because the distance is correlated to the error rate of a classifier via the single feature. This is explained in the theoretical analysis. Next, we guide the RBM to absorb the information from the labels using information geometry by computing the distance, while the traditional algorithm for training the RBM does not take the knowledge about the information from the labels into account. In more detail, through weighting, we enhance the representation of the important features using the RBM. Our algorithm refers to Deep Belief Networks with Information Geometry (IGDBN). Our proposal is able to tackle the aforementioned challenges and results in better performance than the previous methods.

The remainder of this paper is organized as follows. Section II presents our semi-supervised learning method IGDBN in details. Section III presents the experimental results. We conclude the paper in Section IV.

II. MATERIALS AND METHODS

Our method is called IGDBN. We describe its novel procedure for training a DBN and classifying sentiment as follows. First, we preprocess the sentiment classification data set. Second, we evaluate the importance of each feature in the classification from the view of information geometry. Third, in RBMs, we enhance the representation of important features. Fourth, we obtain the DBN that consists of the RBMs and fine-tune the DBN. Finally, the deep network is applied for sentiment classification. The whole procedure is shown in Figure 1.

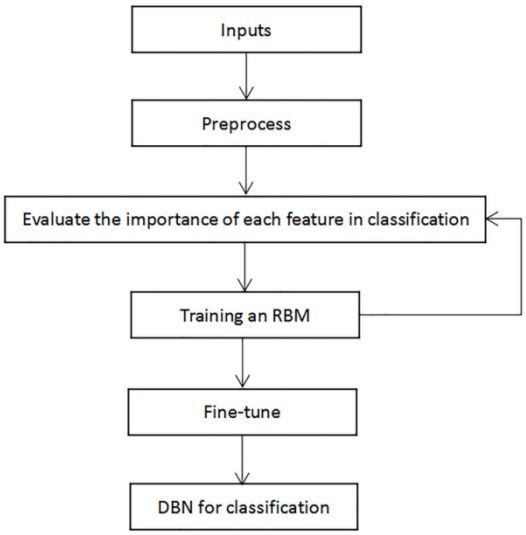

FIGURE 1. The whole procedure for establishing IGDBN and classifying sentiment.

A. PREPROCESS

As the sentiment classification data set is normally composed of many review documents, we need to preprocess them

in advance in the same way as that of [33]. Specifically, we tokenize and downcase each review and represent it as a vector of unigrams, using the frequency as its presence. The details are described as follows.

Each review is represented as a vector x^i. The data set is denoted by

$$X = [x^1, x^2, \ldots, x^{U+T}] \tag{1}$$

where

$$x^i = [x_1^i, x_2^i, \ldots, x_A^i]', \quad i \in 1, 2, \ldots, U+T \tag{2}$$

and where U is the number of training reviews, T is the number of test reviews, A is the number of feature words in the data set, and the frequency of the jth feature word in the ith review is x_j^i for $j \in 1, 2, \ldots, A$.

The L training reviews to be labeled manually are denoted by X^L. Each column of X^L is a vector of a review. These training reviews are chosen randomly. The labels corresponding to L labeled training reviews are aggregated into a set of labels Y. It is denoted as

$$Y = [y^1, y^2, \ldots, y^L] \tag{3}$$

where

$$y^i = [y_1^i, y_2^i]' \tag{4}$$

$$y_c^i = \begin{cases} 1, & if\ x^i \in cth\ class \\ 0, & if\ x^i \notin cth\ class \end{cases} \tag{5}$$

c is the index of classes for $c\epsilon\{1, 2\}$, which corresponds to positive and negative, respectively. If a review x^i is positive, $y^i = [1, 0]$', and otherwise $y^i = [0, 1]$'.

We construct a DBN with one input layer, one output layer and N-1 hidden layers. The input layer h^0 has A units and the output layer h^N has 2 units corresponding to the positive and negative. The output layer has a linear activation function, and every hidden layer uses a sigmoid function as its activation function.

B. TRAINING RBMs

We build the DBN layer by layer using RBMs [34]. Each RBM consists of an input layer and an output layer [35]. And all elements of the inputs and outputs range from [0,1]. All of the training reviews are used as the inputs for the first RBM. And its outputs will be used as the inputs for the next RBM. These will be repeated until the N-1th RBM. An element of the input vector refers to a feature. For each RBM, we study the features to guide its learning.

We divide the labeled data X^L into a positive set X_1^L and a negative set X_2^L. Next, we model the distribution over $X_1^L(j)$ and $X_2^L(j)$, for $j\epsilon\{1, 2, \ldots, A\}$, while $X_c^L(j)$ represents the jth row of X_c^L for $c\epsilon\{1, 2\}$. Each row of X_c^L is associated with a feature as discussed in Section A. It states that the averages of the random variables drawn from the independent distributions converge in distribution to the Gaussian distribution when the number of samples is sufficiently large. Thus, we assume that the distribution over $X_c^L(j)$ satisfies the Gaussian distribution

$$p(x_j|\theta^c(j)) = \frac{1}{\sqrt{2\pi}\sigma_c(j)} \exp(\frac{-|x_j - \mu_c(j)|^2}{2(\sigma_c(j))^2}) \tag{6}$$

where $\theta^c(j) = (\theta_1^c(j), \theta_2^c(j))' = (\mu_c(j), \sigma_c(j))'$. x_j is a random variable. Its mean is $\mu_c(j)$, and its standard deviation is $\sigma_c(j)$. The proposed method can address any other distributions. We will prove it by Theorem 1 in the next section.

To describe the distribution over the jth feature of the samples in set X_c^L, we only need to estimate the mean μ_c and standard deviation σ_c using the sample mean and sample standard deviation as

$$\tilde{\mu}_c(j) = \frac{1}{L_c} \sum_{i=1}^{L_c} x_j^i \tag{7}$$

$$\tilde{\sigma}_c(j) = \sqrt{\frac{1}{L_c - 1} \sum_{i=1}^{L_c} (x_j^i - \tilde{\mu}_c(j))^2} \tag{8}$$

where class c has L_c labeled data for $c\epsilon\{1, 2\}$, j is the index of features, and x_j^i corresponds to the jth feature of the ith labeled data vector. We denote the results as

$$\theta^1(j) = (\mu_1(j), \sigma_1(j))' \tag{9}$$

$$\theta^2(j) = (\mu_2(j), \sigma_2(j))' \tag{10}$$

for each feature indexed by j.

Define

$$H = \{(\mu, \sigma) \in R^2 | \sigma > 0\}. \tag{11}$$

as a half plane.

Let the Fisher Information Matrix (FIM) be

$$G(\theta) = [g_{qz}(\theta)] \tag{12}$$

where q and z are the indexes of the elements in the matrix $G(\theta)$. The element $g_{qz}(\theta)$ is calculated by

$$g_{qz}(\theta) = E\{\frac{\partial \log p(x|\theta)}{\partial \theta_q} \cdot \frac{\partial \log p(x|\theta)}{\partial \theta_z}\} \tag{13}$$

where E signifies the expectation. Then

$$G = (g_{qz}(\mu, \sigma)) = \begin{pmatrix} 1/\sigma^2 & 0 \\ 0 & 2/\sigma^2 \end{pmatrix} \tag{14}$$

We consider the couple (H, G) as a Riemannian manifold. Then, the expression for the metric of (H, G) is

$$ds_F^2 = d\theta^T G(\theta) d\theta = \frac{d\mu^2 + 2d\sigma^2}{\sigma^2} \tag{15}$$

Consider a curve $\theta(t)$ that joins $\theta^1 = \theta(t_1)$ and $\theta^2 = \theta(t_2)$, in which $t_1 \leq t \leq t_2$. Then, the distance along the curve between its endpoints, namely, the two distributions $p(x|\theta^1)$ and $p(x|\theta^2)$, along $\theta(t)$ [36] is defined by

$$D(\theta^1, \theta^2) := \int_{t_1}^{t_2} (\sqrt{(\frac{d\theta}{dt})^T G(\theta)(\frac{d\theta}{dt})}) dt \tag{16}$$

where ":=" stands for "defined as". This distance is dependent on the choice of the curve. The distance between

$p(x|\theta^1)$ and $p(x|\theta^2)$ is defined as the minimum of such distances over all possible curves. The Integrated Fisher Information Distance (IFID) between the two distributions $p(x|\theta^1)$ and $p(x|\theta^2)$ is defined as the integral along the curve $\theta(t)$ that minimizes (16) [37]. That is,

$$D_F(\theta^1,\theta^2) := \min_{\{\theta(t):\theta(t_1)=\theta^1,\theta(t_2)=\theta^2\}} \int_{t_1}^{t_2} \left(\sqrt{\left(\frac{d\theta}{dt}\right)^T G(\theta)\left(\frac{d\theta}{dt}\right)}\right)dt \quad (17)$$

Using the local coordinates, the geodesic equations are given by the Euler–Lagrange equations as

$$\frac{d^2\theta_k}{dt^2} + \sum_{q=1}^{n}\sum_{z=1}^{n}\left(\frac{1}{2}\sum_{q=1}^{n} g^{kl}\left(\frac{\partial g_{ql}}{\partial\theta_z} + \frac{\partial g_{zl}}{\partial\theta_q} - \frac{\partial g_{qz}}{\partial\theta_l}\right)\right)\frac{d\theta_q}{dt}\frac{d\theta_z}{dt} = 0, \quad \forall k,l \in \{1,\ldots,n\} \quad (18)$$

where $\theta = (\theta_1, \theta_2, \ldots, \theta_n)'$.

Any curve $\gamma(t)$ that satisfies (17) is called a geodesic line. Then, for any t, there exists $\varepsilon > 0$, and $\gamma(t)$ is a distance line for $(t-\varepsilon, t+\varepsilon)$. In other words, for any $t_1, t_2 \in (t-\varepsilon, t+\varepsilon)$, the minimal distance line between $\gamma(t_1)$ and $\gamma(t_2)$ is $\gamma(t)$, $t_1 \le t \le t_2$.

By substituting (14) to (17), the distance D_F between $\theta^1(j) = (\mu_1(j), \sigma_1(j))'$ and $\theta^2(j) = (\mu_2(j), \sigma_2(j))'$ in the upper half-plane can be solved as follows:

$$\begin{aligned} &D_F((\mu_1,\sigma_1),(\mu_2,\sigma_2)) \\ &= \sqrt{2}\ln\frac{\left(\left|\left(\frac{\mu_1-\mu_2}{\sqrt{2}}, \sigma_1+\sigma_2\right)\right| + \left|\left(\frac{\mu_1-\mu_2}{\sqrt{2}}, \sigma_1-\sigma_2\right)\right|\right)}{4\sigma_1\sigma_2} \\ &= \sqrt{2}\ln\left(\frac{F((\mu_1,\sigma_1),(\mu_2,\sigma_2)) + (\mu_1-\mu_2)^2 + 2(\sigma_1^2+\sigma_2^2)}{4\sigma_1\sigma_2}\right) \end{aligned} \quad (19)$$

where $|\cdot|$ stands by the standard vector norm in European Space. In addition,

$$\begin{aligned} &F((\mu_1,\sigma_1),(\mu_2,\sigma_2)) \\ &= \sqrt{((\mu_1-\mu_2)^2+2(\sigma_1-\sigma_2)^2)((\mu_1-\mu_2)^2+2(\sigma_1+\sigma_2)^2)} \end{aligned} \quad (20)$$

In (19) and (20), we use μ_1, σ_1, μ_2 and σ_2 instead of $\mu_1(j)$, $\sigma_1(j)$, $\mu_2(j)$ and $\sigma_2(j)$ for simplicity.

Then we substitute (9) and (10) into (19) and (20) to achieve a diagonal matrix $M_{D_F}(j,j) \in R^{A\times A}$. Its element $M_{D_F}(j,j)$ is computed as

$$M_{D_F}(j,j) = 1 - \lambda\frac{1}{1+fun(D_F((\tilde{\mu}_1(j),\tilde{\sigma}_1(j)),(\tilde{\mu}_2(j),\tilde{\sigma}_2(j))))} \quad (21)$$

where fun() denotes a normalization function that divides D_F by the largest one, and λ is a constant that ranges from 0 to 1. $M_{D_F}(j,j)$ ranges from 0 to 1. The IFID is correlated to the error rate. If the distance between the distributions of a feature is larger, then the feature is more discriminative (important), not the other way around. The theoretical analysis is described as follows.

Suppose that a Bayes classifier [38] (which has the lowest error rate) assigns a sample x^i to the cth class according to the value of its jth feature. The rule for classification is as follows:

If $p(c|x_j^i) = \max_{c\in\{1,2\}} p(c|x_j^i)$, then $x^i \in c$th class.

Then, the error rate is defined as

$$p(e) = \int p(e|x_j^i)p(x_j^i)dx_j^i$$

where

$$p(e|x_j^i) = \min_{c\in\{1,2\}}[p(c|x_j^i)] \le \sqrt{p(c=1|x_j^i)p(c=2|x_j^i)}$$

Then,

$$\begin{aligned} p(e) &= \int p(e|x_j^i)p(x_j^i)dx_j^i \\ &\le \int \sqrt{p(c=1|x_j^i)p(c=2|x_j^i)}p(x_j^i)dx_j^i \\ &= \sqrt{p(c=1)p(c=2)}\int \sqrt{p(x_j^i|c=1)p(x_j^i|c=2)}dx_j^i \end{aligned}$$

in which

$$\rho(p(x_j^i|c=1), p(x_j^i|c=2)) = \int \sqrt{p(x_j^i|c=1), p(x_j^i|c=2)}dx_j^i$$

is the Bhattacharyya coefficient[39].

For distributions belonging to the same exponential family (e.g., $p(x_j^i|c=1)$ and $p(x_j^i|c=2)$ are Gaussians), we have

$$\rho = e^{-J_F(\theta^1,\theta^2)}$$

where J_F is a Jensen divergence defined over the natural parameter space.

To make the bound for $P(e)$ tighter, we may consider for $\alpha \in [0,1]$ that

$$\begin{aligned} &\rho(p(x_j^i|c=1), p(x_j^i|c=2)) \\ &\quad \le \rho_\alpha(p(x_j^i|c=1), p(x_j^i|c=2)) \\ &\quad = \int [p(x_j^i|c=1)]^\alpha [p(x_j^i|c=2)]^{1-\alpha}dx_j^i \end{aligned}$$

To scale it,we have[40]

$$D_\alpha = \frac{1-\rho_\alpha}{\alpha(1-\alpha)}$$

It is the IFID in information geometry when we set $\alpha \in [-\frac{1}{2}, \frac{1}{2}]$ instead of [0,1] by remapping $\alpha \leftarrow \alpha - \frac{1}{2}$. Thus, when the IFID is larger, ρ_α is smaller. It makes the upper bound of the error rate of the classification according to the jth feature lower. Apart from the IFID, there are many other alternative measures, among which a popular one is the Kullback-Leibler divergence (KLD) [41]. However, the KLD cannot reflect the infinitesimal difference between two distributions as the IFID can. The proof this can be found in the Appendix.

Therefore, the value of $M_{D_F}(j,j)$ reflects the importance of the jth feature. We call it the importance factor.

Next, we will show that all the calculations discussed above can be extended to address any other distribution.

Theorem 1: Let $k^1, k^2, \ldots, k^j$ be real numbers that follow the same distribution. The mean and standard deviation of such numbers are μ_1 and σ_1, respectively. Let

$$\mu_2 = (\sum_{i=1}^{j} k^i)/j$$

$$\sigma_2 = \sqrt{[\sum_{i=1}^{j} (k^i - \mu_2)^2]/(j-1)}$$

Then, for any $0 < \varepsilon < 1$, the assertion

$$\lim_{j->\infty} D_F((\mu_1,\sigma_1),(\mu_2,\sigma_2)) = 0$$

$$\lim_{j->\infty} D_F((\mu_1,\sigma_1),(\mu_2,\sigma_2))/\sqrt{\mu^2+\sigma^2} > 0$$

holds with atleast a probability of $1-\varepsilon$ where

$$\mu = (\mu_2 - \mu_1)/\sigma_1$$
$$\sigma = \sigma_2/\sigma_1 - 1$$

The proof can be seen in the Appendix. According to Theorem 1, all the calculations discussed above can be extended to address any other distribution. This is because when j is large enough, the D_F between a Gaussian distribution and another distribution trends to zero, and thus the distribution will converge to the Gaussian distribution.

The feature, which has a larger importance factor, usually plays a more important role in classification. To enhance the representation of important features by RBMs, it is reasonable to promote the amplitudes of the weights of the connections from the visible units corresponding to these important features. At the same time, the weights of useless features should be restrained. Thus, we redefine the energy as

$$E(v,h) = -b^T M_{D_F} v - r^T h - h^T W M_{D_F} v \tag{22}$$

where a visible vector v of dimension A and a layer h of B binary hidden units. The parameters of the RBM are denoted by b, r and W. Each element of v represents a feature that is multiplied by its importance factor in M_{D_F}.

$g()$ is the logistic sigmoid function defined as

$$g(t) = \frac{1}{1+e^{-t}} \tag{23}$$

Given a random training example, v, the probability that the binary state, h_o, of each hidden unit is 1 will be expressed as

$$p(h_o|v) = g(r_o + \sum_j w_{jo} M_{D_F}(j,j) v_j) \tag{24}$$

We begin from a random state of visible units where a single step of Gibbs sampling determines the hidden units' state using (24) and then computes the visible units' state using (25).

$$p(v_j|h) = g(M_{D_F}(j,j) b_j + \sum_o w_{jo} M_{D_F}(j,j) h_o) \tag{25}$$

The parameter updates require performing the 1-step Contrastive Divergence [42].

$$\Delta w_{jo} = \alpha(< M_{D_F}(j,j) \times v_j h_o >_{data} - < M_{D_F}(j,j) \times v_j h_o >_{recon}) \tag{26}$$

$$\Delta b_j = \alpha(< M_{D_F}(j,j) \times v_j >_{data} - < M_{D_F}(j,j) \times v_j >_{recon}) \tag{27}$$

$$\Delta r_o = \alpha(< h_o >_{data} - < h_o >_{recon}) \tag{28}$$

where α is the learning rate, $< \cdot >_{data}$ denotes an expectation with respect to the data distribution and $< \cdot >_{recon}$ is the corresponding expectation when the features are being driven by the reconstructed counts.

Each RBM is associated with a hidden layer in the DBN. The activation of hidden units in one RBM is treated as the training data for its next RBM, and the labels are inherited. For the latter RBM, we repeat the process described above. Finally, we achieve N-1 RBMs.

C. FINE-TUNE AND CLASSIFICATION

We refine the parameter space using L labeled reviews by back-propagation. In this task, we define the optimization problem as

$$\underset{W}{\operatorname{argmin}} f(h^N(X^L, Y^L)) \tag{29}$$

$$f(h^N(X^L), Y^L) = \frac{1}{2} \sum_{i=1}^{L} \sum_{c=1}^{2} (h_c^N(x^i) - y_c^i)^2 \tag{30}$$

where c is the index of classes and h^N is the activation function of the output layer.

The label of new data is denoted by $\hat{c}$. It is determined by

$$\hat{c} = \underset{c}{\operatorname{argmax}}\, h_c^N(x) \tag{31}$$

III. EXPERIMENTAL ANALYSIS

A. EXPERIMENTAL SETUP

We use five sentiment classification data sets from the UCI Machine Learning Repository, as has been done in previously published works. The data sets include electronics (ELE), kitchen appliances (KIT), movies (MOV), books (BOO) and DVDs (DVD). Each of them contains 1000 positive and 1000 negative reviews. We divide the 2000 reviews into two parts. Half of the reviews are randomly selected as training data and the remaining reviews are used for testing. All algorithms are tested with cross-validation for 10 rounds. Then, the average of these 10 experiment results is reported. The percent of labeled training samples ranges from 20% to 80%. We adapt the structures of the deep networks in [43]. We also set the same parameters for all neural networks. The learning rate is 0.05, the momentum is 0, the number of iterations is 2000, and λ is set to 0.1 based on experience.

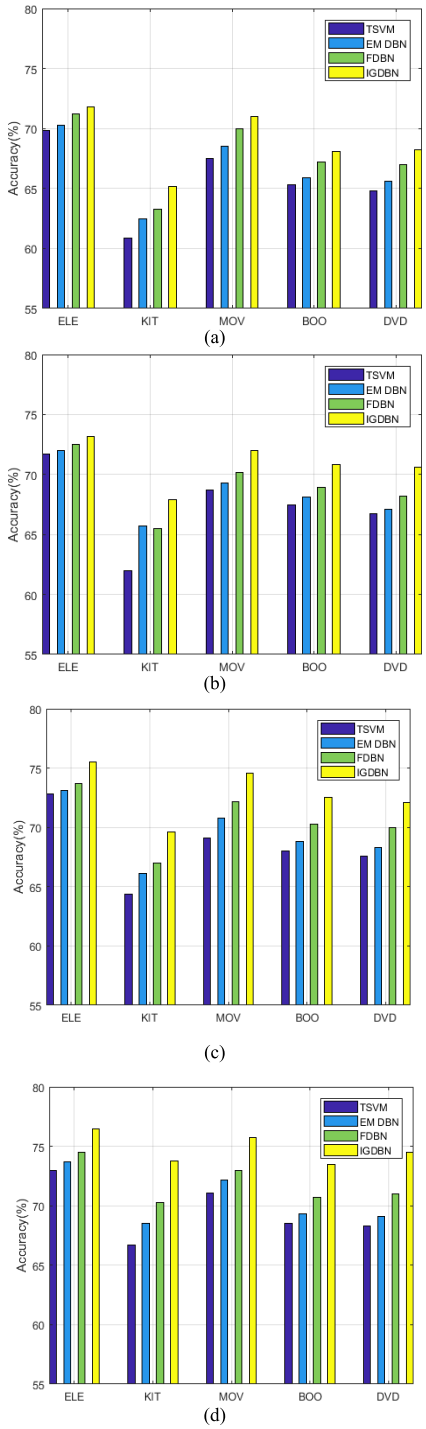

FIGURE 2. Test accuracy on five data sets for TSVM, FDBN, EM DBN and IGDBN. (a). Test accuracy on five data sets for TSVM, FDBN, EM DBN and IGDBN when the percent of labeled training samples is set to 20%. (b). Test accuracy on five data sets for TSVM, FDBN, EM DBN and IGDBN when the percent of labeled training samples is set to 40%. (c). Test accuracy on five data sets for TSVM, FDBN, EM DBN and IGDBN when the percent of labeled training samples is set to 60%. (d). Test accuracy on five data sets for TSVM, FDBN, EM DBN and IGDBN when the percent of labeled training samples is set to 80%.

B. PERFORMANCE COMPARISON

We compare the classification performance of IGDBN with two representative semi-supervised learning classifiers, i.e., the Transductive SVM (TSVM) [44] and Fuzzy Deep Belief Networks (FDBN) [43]. In addition, we also apply the EM algorithm [45] to deep belief networks so that we can compare the proposed method with the method using information geometry. For simplicity, we call this kind of DBN an EM DBN.

The test accuracy on the five data sets for TSVM, FDBN, EM DBN and IGDBN (proposed method) can be seen in Figure 2. We can see that the performance of the IGDBN is better than the TSVM, FDBN and EM DBN on all five data sets. This proves the effectiveness of our proposed learning method, which labels the same number of reviews, in that it can obtain better performance than the other semi-supervised methods. Intuitively, the improvement of our method is significant compared with previous methods. The improved

TABLE 1. Improved Percentages on accuracy. (a) Average improved percentages over five data sets when the percent of labeled training samples is set to 20%. (b) Average improved percentages over five data sets when the percent of labeled training samples is set to 40%. (c) Average improved percentages over five data sets when the percent of labeled training samples is set to 60%. (d) Average improved percentages over five data sets when the percent of labeled training samples is set to 80%.

%	TSVM	EM DBN	FDBN	IGDBN
TSVM	-	1.03	2.1	3.27
EM DBN		-	1.07	2.23
FDBN			-	1.17
IGDBN				-

(a)

%	TSVM	EM DBN	FDBN	IGDBN
TSVM	-	1.53	1.93	3.57
EM DBN		-	0.4	1.47
FDBN			-	2.03
IGDBN				-

(b)

%	TSVM	EM DBN	FDBN	IGDBN
TSVM	-	1.23	2.37	2.03
EM DBN		-	0.8	3.23
FDBN			-	2.42
IGDBN				-

(c)

%	TSVM	EM DBN	FDBN	IGDBN
TSVM	-	1.2	2.33	5.06
EM DBN		-	1.13	3.87
FDBN			-	2.73
IGDBN				-

(d)

$$D_F((\mu_1,\sigma_1),(\mu_2,\sigma_2)) = \sqrt{2}\ln\{[F((\mu_1,\sigma_1),(\mu_2,\sigma_2)) + (\mu_1-\mu_2)^2 + 2(\sigma_1^2+\sigma_2^2)]/4\sigma_1\sigma_2\}$$
$$= \sqrt{2}\ln[\frac{\sigma_1^2\sqrt{(\mu^2+2\sigma^2)(\mu^2+8+O(\sigma))} + \sigma_1^2\mu^2 + 4\sigma_1^2(1+\sigma+O(\sigma^2))}{4\sigma_1\sigma_2}]$$

percentages on accuracy are given in Table I. Under different percentages of labeled training samples, IGDBN outperforms the previous algorithms constantly. The improvement of IGDBN over FDBN and EM DBN is comparable to the improvement of those two methods made over previous ones. While the percent of labeled training samples becomes larger, the improvement of the IGDBN tends to be more prominent. That is because more samples lead to equations (7) and (8) having enough samples to approach the real mean and standard deviation.

IV. CONCLUSION

According to the shortcomings of the current methods for sentiment classification, we propose a sophisticated algorithm based on deep learning and information geometry. The novel semi-supervised learning algorithm IGDBN addresses the sentiment classification problem with a number of labeled reviews. The experiments show that the proposed method has advantages in accuracy compared with the previous algorithms.

APPENDIX

PROOF FOR THEOREM 1

By the Central Limit Theorem, for any distribution, for a sufficiently large j, we have

$$(\mu_2-\mu_1)/\sigma_1 \sim N(0,\ 1/j)$$
$$(\sigma_2)^2/(\sigma_1)^2 \sim N(1,\ 1/(j-1))$$

Then, there exists a positive function $c(j)$, which is decreasing with zero as the limit, such that

$$p\{|\mu| \le c(j)\} > 1-\varepsilon$$
$$p\{|\sigma| \le c(j)\} > 1-\varepsilon$$

For a large j, we deduce that $D_F((\mu_1,\sigma_1),(\mu_2,\sigma_2))$, as shown at the top of this page.

Then,

$$\sqrt{2}\ln[c_1(r+o(r))+1] \le D_F((\mu_1,\sigma_1),(\mu_2,\sigma_2)) \le \sqrt{2}\ln[c_2(r+o(r))+1]$$

where $r = \sqrt{\mu^2+\sigma^2}$, and c_1 and c_2 are positive constants. The results hold true.

Next, we prove the superiority of D_F compared with KLD.

The symmetric form of KLD is[41]

$$KLD((\mu_1,\sigma_1)||(\mu_2,\sigma_2)) = \frac{1}{2}[2\ln(\sigma_2/\sigma_1) + \sigma_1^2/\sigma_2^2 + (\mu_1-\mu_2)^2/\sigma_2^2 - 1]$$

Then, for a large j, we have

$$KLD((\mu_1,\sigma_1)||(\mu_2,\sigma_2)) \le o(\sqrt{\mu^2+\sigma^2})$$

Then, according to Theorem 1, with at least a probability of $1-\varepsilon$,

$$\lim_{n->\infty} KLD((\mu_1,\sigma_1)||(\mu_2,\sigma_2))/\sqrt{\mu^2+\sigma^2} = 0$$

which implies $KLD(\cdot)$ has lower sensitivity than $D_F(\cdot)$.

REFERENCES

[1] L. Shoushan, S. Y. M. Lee, Y. Chen, C. Huang, and G. Zhou, "Sentiment classification and polarity shifting," in *Proc. 23rd Int. Conf. Comput. Linguistics*, Beijing, China, 2010, pp. 635–643.
[2] X. Wan, "Bilingual co-training for sentiment classification of Chinese product reviews," *Comput. Linguistics*, vol. 37, no. 3, pp. 587–616, Jan. 2011.
[3] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up? Sentiment classification using machine learning techniques," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, Philadelphia, PA, USA, 2002, pp. 79–86.
[4] P. D. Turney, "Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews," in *Proc. 40th Annu. Meeting Assoc. Comput. Linguistics*, Philadelphia, PA, USA, 2002, pp. 417–424.
[5] H. He, Z. Li, C. Yao, and W. Zhang, "Sentiment classification technology based on Markov logic networks," *New Rev. Hypermedia Multimedia*, vol. 22, no. 3, pp. 243–256, Jul. 2016.
[6] L. Zhuang, F. Jing, and X.-Y. Zhu, "Movie review mining and summarization," in *Proc. 15th ACM Int. Conf. Inform. Knowl. Manage.*, Arlington, VA, USA, 2006, pp. 43–50.
[7] A. Go, R. Bhayani, and L. Huang, "Twitter sentiment classification using distant supervision," Stanford Univ., Stanford, CA, USA, Project Rep. CS224N, Nov. 2009.
[8] F. Wu, Y. Song, and Y. Huang, "Microblog sentiment classification with contextual knowledge regularization," in *Proc. 29th AAAI Conf. Artif. Intell.*, Austin, TX, USA, 2015, pp. 2332–2338.
[9] R. McDonald, K. Hannan, T. Neylon, M. Wells, and J. Reynar, "Structured models for fine-to-coarse sentiment analysis," in *Proc. Annu. Meeting Assoc. Comput. Linguistics*, Prague, Czech Republic, 2007, pp. 432–439.
[10] N. Zainuddin, A. Selamat, and R. Ibrahim, "Hybrid sentiment classification on twitter aspect-based sentiment analysis," *Appl. Intell.*, vol. 48, no. 5, pp. 1218–1232, May 2018.
[11] V. García-Díaz, J. P. Espada, R. G. Crespo, B. C. P. G-Bustelo, and J. M. C. Lovelle, "An approach to improve the accuracy of probabilistic classifiers for decision support systems in sentiment analysis," *Appl. Soft Comput.*, vol. 67, pp. 822–833, Jun. 2018.
[12] A. I. Pratiwi and K. Adiwijaya, "On the feature selection and classification based on information gain for document sentiment analysis," *Appl. Comput. Intell. Soft Comput.*, vol. 2018, pp. 1407817-1–1407817-5, Feb. 2018.
[13] A. Aue and M. Gamon, "Customizing sentiment classifiers to new domains: A case study," in *Proc. Int. Conf. Recent Adv. Natural Lang. Process.*, Borovets, Bulgaria, 2005, pp. 33–45.
[14] S. Tan, G. Wu, H. Tang, and X. Cheng, "A novel scheme for domain-transfer problem in the context of sentiment analysis," in *Proc. 16th CIKM*, New York, NY, USA, 2007, pp. 979–982.
[15] S. Li and C. Zong, "Multi-domain sentiment classification," in *Proc. 48th Annu. Meeting Assoc. Comput. Linguistics Human Lang. Technol.*, Columbus, OH, USA, 2008, pp. 257–260.
[16] S. J. Pan, X. Ni, J.-T. Sun, Q. Yang, and Z. Chen, "Cross-domain sentiment classification via spectral feature alignment," in *Proc. Int. Conf. World Wide Web*, Raleigh, NC, USA, 2010, pp. 751–760.

[17] S. Li, C.-R. Huang, G. Zhou, and S. Y. M. Lee, "Employing personal/impersonal views in supervised and semi-supervised sentiment classification," in *Proc. 48th Annu. Meeting Assoc. Comput. Linguistics*, Uppsala, Sweden, 2010, pp. 414–423.
[18] P. D. Turney and M. L. Littman, "Measuring praise and criticism: Inference of semantic orientation from association," *ACM Trans. Inform. Syst.*, vol. 21, no. 4, pp. 315–346, Oct. 2003.
[19] J. Read and J. Carroll, "Weakly supervised techniques for domain-independent sentiment classification," in *Proc. Int. CIKM Workshop Topic-Sentiment Analy. Mass Opinion*, Hong Kong, 2009, pp. 45–52.
[20] M. S. Hajmohammadi, R. Ibrahim, A. Selamat, and H. Fujita, "Combination of active learning and self-training for cross-lingual sentiment classification with density analysis of unlabelled samples," *Inf. Sci.*, vol. 317, pp. 67–77, Oct. 2015.
[21] N. F. F. da Silva, L. F. S. Coletta, E. R. Hruschka, and E. R. Hruschka, Jr., "Using unsupervised information to improve semi-supervised tweet sentiment classification," *Inf. Sci.*, vols. 355–356, pp. 348–365, Aug. 2016.
[22] V. Sindhwani and P. Melville, "Document-word co-regularization for semi-supervised sentiment analysis," in *Proc. 8th IEEE Int. Conf. Data Mining*, Pisa, Italy, Dec. 2008, pp. 1025–1030.
[23] M. S. Hajmohammadi, R. Ibrahim, and A. Selamat, "Cross-lingual sentiment classification using multiple source languages in multi-view semi-supervised learning," *Eng. Appl. Artif. Intell.*, vol. 36, pp. 195–203, Nov. 2014.
[24] S. Zhou, Q. Chen, and X. Wang, "Active deep learning method for semi-supervised sentiment classification," in *Proc. 23rd Int. Conf. Comput. Linguistics, Posters*, Beijing, China, 2010, pp. 1515–1523.
[25] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, 2006.
[26] Y. Fang, H. Tan, and J. Zhang, "Multi-strategy sentiment analysis of consumer reviews based on semantic fuzziness," *IEEE Access*, vol. 6, pp. 20625–20631, Apr. 2018.
[27] R. E. Kass and P. W. Vos, *Geometrical Foundations of Asymptotic Inference*. New York, NY, USA: Wiley, 1997.
[28] S.-I. Amari and M. Kawanabe, "Information geometry of estimating functions in semi-parametric statistical models," *Bernoulli*, vol. 3, no. 1, pp. 29–54, Mar. 1997.
[29] S. Amari, "Information geometry of statistical inference—An overview," in *Proc. IEEE Inf. Theory Workshop*, Bengaluru, India, Oct. 2002, pp. 86–89.
[30] M. Basseville, "Divergence measures for statistical data processing—An annotated bibliography," *Signal Process.*, vol. 93, no. 4, pp. 621–633, Apr. 2013.
[31] S.-I. Amari, "Information geometry in optimization, machine learning and statistical inference," *Frontiers Elect. Electron. Eng. China*, vol. 5, no. 3, pp. 241–260, Sep. 2010.
[32] S.-I. Amari, "Information geometry and its applications: Convex function and dually flat manifold," in *LIX Fall Colloquium on Emerging Trends in Visual Computing*, F. Nielsen, Ed. Berlin, Germany: Springer, 2009, pp. 75–102.
[33] S. Dasgupta and V. Ng, "Mine the easy, classify the hard: A semi-supervised approach to automatic sentiment classification," in *Proc. Joint Conf. 47th Annu. Meeting ACL, 4th Int. Joint Conf. Natural Lang. Process. (IJCNLP)*, Singapore, vol. 2, 2009, pp. 701–709.
[34] Y. Bengio, "Learning deep architectures for AI," *Found. Trends Mach. Learn.*, vol. 2, no. 1, pp. 1–127, Jan. 2009.
[35] P. Smolensky, "Information processing in dynamical systems: Foundations of harmony theory," in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, vol. 1, D. E. Rumelhart, J. L. McClelland, and PDP Research Group, Eds. Cambridge, MA, USA: MIT Press, 1986, pp. 194–281.
[36] M. L. Menéndez, D. Morales, L. Pardo, and M. Salicrú, "Statistical tests based on geodesic distances," *Appl. Math. Lett.*, vol. 8, no. 1, pp. 65–69, Jan. 1995.
[37] K. M. Carter, R. Raich, W. G. Finn, and A. O. Hero, III, "FINE: Fisher information nonparametric embedding," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 11, pp. 2093–2098, Nov. 2009.
[38] L. Devroye, L. Györfi, and G. Lugosi, *A Probabilistic Theory of Pattern Recognition*. New York, NY, USA: Springer, 1996.
[39] F. Nielsen and V. Garcia. (2009). "Statistical exponential families: A digest with flash cards." [Online]. Available: https://arxiv.org/abs/0911.4863
[40] F. Nielsen, "Pattern learning and recognition on statistical manifolds: An information-geometric review," in *Similarity-Based Pattern Recognition*. Berlin, Germany: Springer, 2013, pp. 1–25.
[41] S. Kullback and R. A. Leibler, "On information and sufficiency," *Ann. Math. Statist.*, vol. 22, no. 1, pp. 79–86, Mar. 1951.
[42] G. E. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural Comput.*, vol. 14, no. 8, pp. 1771–1800, Aug. 2002.
[43] S. Zhou, Q. Chen, and X. Wang, "Fuzzy deep belief networks for semi-supervised sentiment classification," *Neurocomputing*, vol. 131, pp. 312–322, May 2014.
[44] S. Kamvar, D. Klein, and C. Manning, "Spectral learning," in *Proc. Int. Joint Conf. Artif. Intell.*, Catalonia, Spain, 2003, pp. 561–566.
[45] S.-I. Amari, "Information geometry of the EM and EM algorithms for neural networks," *Neural Netw.*, vol. 8, no. 9, pp. 1379–1408, Jan. 1995.

MENG WANG is currently pursuing the Ph.D. degree with the Faculty of Information Technology, Beijing University of Technology. Her research interests are in the areas of signal processing and artificial intelligence.

In 2014, she participated in the study of case-based reasoning systems for judgment. In 2015, she participated in the research on several problems of dynamic scheduling for concurrent multiple DAGs in a hybrid cloud-computing environment.

ZHEN-HU NING received the Ph.D. degree in computer science from the Beijing University of Technology in 2016. He is currently a Lecturer with the Faculty of Information Technology, Beijing University of Technology. His research interests are in the areas of information safety and artificial intelligence.

CHUANGBAI XIAO received the Ph.D. degree from Tsinghua University in 1995. Since 2001, he has been teaching and researching with the Faculty of Information Technology, Beijing University of Technology, where he is currently a Professor. He has authored or co-authored over 100 papers in peer-reviewed journals, conferences, or workshops.

TONG LI received the Ph.D. degree in computer science from the Beijing University of Technology in 2016. He is currently a Lecturer with the Faculty of Information Technology, Beijing University of Technology. His research interests are in the areas of information safety and artificial intelligence.