Multimedia Tools and Applications (2019) 78:20637–20653
https://doi.org/10.1007/s11042-019-7427-5

Joint sentiment/topic modeling on text data using a boosted restricted Boltzmann Machine

Masoud Fatemi[1] · Mehran Safayani[1]

Received: 27 July 2018 / Revised: 25 January 2019 / Accepted: 25 February 2019 /
Published online: 5 March 2019

Abstract

Recently by the development of the Internet and the Web, different types of social media such as web blogs become an immense source of text data. Through the processing of these data, it is possible to discover practical information about different topics, individual's opinions and a thorough understanding of the society. Therefore, applying models which can automatically extract the subjective information from documents would be efficient and helpful. Topic modeling methods and sentiment analysis are the raised topics in natural language processing and text mining fields. In this paper a new structure for joint sentiment-topic modeling based on a Restricted Boltzmann Machine (RBM) which is a type of neural networks is proposed. By modifying the structure of RBM as well as appending a layer which is analogous to sentiment of text data to it, we propose a generative structure for joint sentiment topic modeling based on neural networks. The proposed method is supervised and trained by the Contrastive Divergence algorithm. The new attached layer in the proposed model is a layer with the multinomial probability distribution which can be used in text data sentiment classification or any other supervised application. The proposed model is compared with existing models in the experiments such as evaluating as a generative model, sentiment classification, information retrieval and the corresponding results demonstrate the efficiency of the method.

Keywords Topic modeling · Sentiment analysis · Neural networks · Restricted Boltzmann Machine · Probabilistic model

1 Introduction

Nowadays the ultimate objective of Artificial Intelligence is to provide a way to perform different activities of humans automatically and as quickly as possible. With the rapid

✉ Mehran Safayani
safayani@cc.iut.ac.ir

Masoud Fatemi
m.fatemi@ec.iut.ac.ir

[1] Department of Electrical and Computer Engineering, Isfahan University of Technology,
Isfahan 84156-83111, Iran

expansion of the Internet in recent decades, various types of social media have been transformed into massive sources of data, and especially text data, which can be processed to obtain valuable information about people's viewpoints and general understanding toward different topics [15]. The developments in the fields of text data mining and Natural Language Processing (NLP) have made major contributions to the understanding and analysis of these massive volumes of text data. Nevertheless, there is still a high demand for the methods capable of automatic analysis of massive unstructured data so as to extract the valuable information. Topic models are a class of text analysis methods that have recently drawn major attention from researchers of different fields, especially those working on NLP and text data mining [17].

Topic models consider text documents as a mixture of multiple topics, where each topic can be treated as a probability distribution over words [1, 22]. In the field of text data mining, topic models are those models that can detect and extract an abstract of the topics discussed in one or multiple documents [1, 2]. In other words, topic modeling methods are the tools that allow us to model text documents or any other set of discrete data. The goal of these models is to find a short description of the members of the dataset for effective processing of the original dataset without losing the statistical dependencies necessary for basic tasks such as classification or summarization [2].

In the fields of NLP and text data mining, having a proper perspective about people's manner of thinking is an important part of data collection [19, 20]. The advent and popularization of instant commenting tools such as online review sites and personal blogs has created new opportunities as well as challenges for understanding the people's opinions through information technology [20]. In the field of opinion mining and sentiment analysis, the goal is to design and use a method or tool for automatic identification of conceptual information such as views, attitudes, and sentiments in a text document [15, 19].

While there have been many invaluable researches with major contributions to the above applications, there is a common deficiency in the existing body of literature and that is the concentration on detection of overall sentiment of documents without an in-depth analysis to identify latent topics and their associated sentiments. Each review contains and addresses a set of topics [15]. For example, a review of a restaurant may address topics such as food, service, location, and price, among others. Detection of these topics is a necessary part of the process of retrieving more detailed information, but the absence of a sentiment analysis on the extracted topics may undermine the quality of the results. This is because users are interested not only in the overall sentiment of a review and its topical information, but also to the sentiment or opinion associated with each topic. For example, a customer may be content about the quality and price of food, but not about the service and location. Therefore, simultaneous detection of topics and their associated sentiments (join sentiment/topic modeling) is far more desirable as it provides information that is far more valuable [15]. In addition, detecting the sentiment of documents and topics can be as instrumental in the information retrieval as it is in topic detection in text mining. Hence, the methods of automatic joint sentiment/topic modeling of text documents could be of great scientific and economic value.

The present paper is focused on the processing of text data. Our goal is to use the capabilities of artificial neural networks to determine the distribution of topics discussed in the documents of a database and the word distribution and sentiments associated with each topic. In the text mining literature, the aforementioned process is known as joint sentiment/topic modeling. The good performance of neural networks in topic modeling, especially when compared to previous approaches that use Bayesian structures [7, 11], and the limitations of Bayesian methods such as practical impossibility of exact inference [7], necessitate more attention to the potentials and use of neural networks in this application. The proposed approach is a supervised generative probabilistic model based on the Restricted Boltzmann Machine (RBM) neural network [6, 21] for the joint sentiment/topic modeling of text data. Like other RBM-based methods, the model is trained using the Contrastive Divergence (CD) [4–6, 23] learning algorithm.

The rest of this paper is organized as follows: In the second section, we review the previous work on the estimation of probability distributions in input data, topic modeling, sentiment analysis, and joint sentiment/topic modeling of text data. In the third section, the theoretical foundations and the theory of the proposed model are explained. In this section, we use a well-known model as the basis of work to develop a new model and then describe its various parts and the relationships required in each part. In the fourth section, we explain the steps taken to evaluate the proposed model and then compare its performance in different experiments with other models. Also, we created two new datasets named *Sentiment-20NG*[1] and *MRMDS*[2] to evaluate the proposed model in the field of information retrieval. These two new datasets will be described in this section. In the final section, we present the conclusions of the paper and make some suggestions for improving and further developing the proposed model.

2 Related works

In this section, we review the literature related to topic modeling, estimation of probability distributions in input data, and joint sentiment/topic modeling based on both neural networks and also Bayesian approaches.

The Restricted Boltzmann Machine (RBM) is an unsupervised two-layer neural network for the estimation of distribution of binary input data. This generative probabilistic model was first introduced in 1986 by Smolensky [21] and was later developed by Hinton in 2002 [6]. Inspired by the RBM model, in 2011 Larochelle et al. introduced the Neural Autoregressive Distribution Estimation (NADE) [12], which is an unsupervised generative probabilistic method for modeling the probability of discrete data. NADE eliminated the limitation of RBM in high dimensional joint probability estimations by the use of fully visible Bayes networks for probability calculations. The earliest neural network-based topic model is the Replicated Softmax Model (RS) introduced by Hinton and Salakhutdinov in 2009 [7], which is an extension of the RBM model used to detect the distribution of topics in text data [7]. In 2012, Larochelle and Lauly combined NADE and RS to develop an unsupervised neural network-based topic modeling method called the Document Neural Autoregressive Distribution Estimation (DocNADE) [11].

In the category of Bayesian topic models, the well-known Latent Dirichlet Allocation (LDA) model introduced by Blei et al. in 2003 [2] has long served as the basis of all methods of this category. LDA is a generative probabilistic method in which a text document is considered as a mixed distribution over topics, where each topic is characterized by a distribution over words. Additionally, information retrieval and efficiently searching in a huge and massive amount of data specifically with different modalities, as an application of topic models, have attracted high attention due to its importance. In the category of Bayesian topic models, Yu and Qin in 2016 [14] proposed a correlation topic model for cross-modal

[1] Available at: https://github.com/Masoud-Fatemi/Sentiment-20NG

[2] Available at: https://github.com/Masoud-Fatemi/MRMDS-dataset

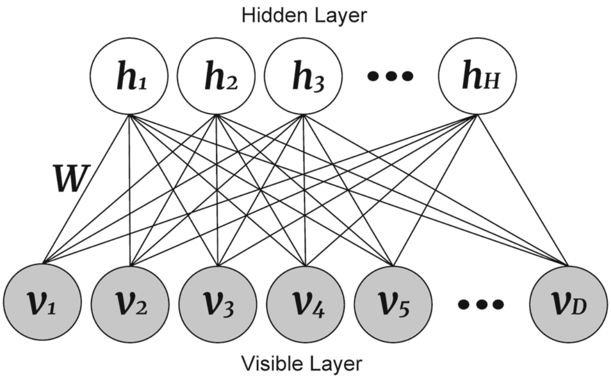


Fig. 1 RBM model

information retrieval which acts upon the combination of text and image. The cross-modality correlation enables this model to calculate the conditional probability of interest in one modality based on a query in another modality. In the field of sentiment analysis and opinion mining, social media has high potential to concentrate on it. In 2016 Tang and Ni proposed a model [16] to deal with the crowd unfollow problem in Twitter. Actually, their proposed model has the ability to detect crowd unfollow in real-time which is a great feature in social network analysis.

All of the aforementioned models are capable of detecting the topics in text data or doing sentiment analysis. There is however another group of topic models that can detect the topics as well as the sentiments associated with each one. This group includes the Aspect-Sentiment Unification Model (ASUM) introduced by Jo and Oh in 2011 [9] for detecting topics and sentiments in online reviews. ASUM is an extension of LDA and falls in the category of generative probabilistic graph models. In 2012, Lin et al. introduced the weakly supervised joint sentiment-topic (JST) detection model [15]. The advantage of JST over its competitors is in its weakly supervised nature, which allows it to be easily adapted for other domains without noticeable decrease in performance.

3 Proposed model

The basis of the proposed model in this paper is the RBM [6] neural network shown in Fig. 1.

In RBM, probability distributions of input data are obtained by minimization of an energy function defined as:

$$E(\mathbf{v},\mathbf{h}) = -\sum_i \sum_j v_i W_{ij} h_j - \sum_i v_i a_i - \sum_j h_j b_j. \tag{1}$$

In (1), $\theta = \{W, \mathbf{a}, \mathbf{b}\}$ is the set of model parameters. $W_{D\times H}$ is the weight matrix for the connections between the input layer and the hidden layer, where D is the size of the input vector, and H is the size of the hidden layer. The parameter $\mathbf{a}$ is the bias vector of the input layer of size D and the parameter $\mathbf{b}$ is the bias vector of the hidden layer of size H.

Now, assume that our aim is to utilize RBM to model the discrete data $\mathbf{v}$ where $\mathbf{v} \in \{1, ..., K\}^D$. Here, K is the dictionary size, D is the document size, and $\mathbf{h} \in \{0, 1\}^H$ is the hidden layer. We assume the matrix $\mathbf{V}$ of size $K \times D$ as the visible binary matrix where

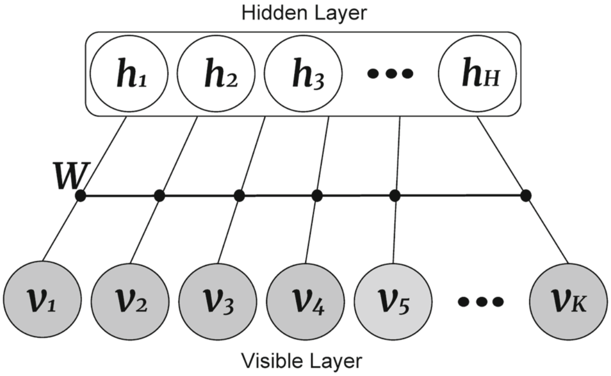


Fig. 2 RS model

$v_{ki} = 1$ if visible unit i takes on the kth value. the energy function for the state $\{\mathbf{V}, \mathbf{h}\}$ is defined as:

$$E(\mathbf{V}, \mathbf{h}) = -\sum_{i=1}^{D}\sum_{j=1}^{H}\sum_{k=1}^{K} W_{kij} h_j v_{ki} - \sum_{i=1}^{D}\sum_{k=1}^{K} v_{ki} a_{ki} - \sum_{j=1}^{H} h_j b_j. \tag{2}$$

It worth to mention that RBM corresponds to (1) assumes that the input data are vectors of size D and the number of hidden layer neurons is H. In this RBM, W which is the weight matrix between the input layer (visible layer) and the hidden layer is of size $D \times H$. On the other hand RBM corresponds to (2) assumes that the input data are matrices of size $K \times D$, therefore it has $K \times D$ neurons in its input layer and the number of hidden layer neurons is H. The weight tensor for this RBM has the size of $D \times H \times K$. Therefore, whenever $K = 1$, i.e., the input data to the model are vectors instead of matrices, the (2) is equal to (1).

The conditional distributions are calculated in the form of softmax and logistic function[7]:

$$p(v_{ki} = 1|\mathbf{h}) = \frac{exp(a_{ki} + \sum_{j=1}^{H} h_j W_{kij})}{\sum_{k=1}^{K} exp(a_{ki} + \sum_{j=1}^{H} h_j W_{kij})} \tag{3}$$

$$p(h_j = 1|\mathbf{V}) = \sigma\left(b_j + \sum_{i=1}^{D}\sum_{k=1}^{K} v_{ki} W_{kij}\right), \tag{4}$$

where $\sigma(x) = 1/(1 + exp(-x))$ is the logistic function (logistic curve).

Now suppose, for each document, we create an independent RBM with as many softmax units as there are words in the document. With the order of words ignored, all of the softmax units can share the weights that connect them to the hidden layer. Therefore, for a document consisting of D words, the energy function for the state $\{\mathbf{V}, \mathbf{h}\}$ is defined as:

$$E(\mathbf{v}, \mathbf{h}) = -\sum_{j=1}^{H}\sum_{k=1}^{K} W_{jk} h_j \hat{v}_k - \sum_{k=1}^{K} v_k a_k - D\sum_{j=1}^{H} h_j b_j \tag{5}$$

where $\hat{v}^k = \sum_{i=1}^{D} v_{ik}$. Equation (5) is in fact the RS model proposed by Hinton and Salakhutdinov [7] which is shown in Fig. 2.

The formulation (5) constitutes the basis of the proposed structure. The model of this paper is developed by extending the above formulations as described below. The proposed model is a RBM-based generative probabilistic model for sentiment/topic modeling of text

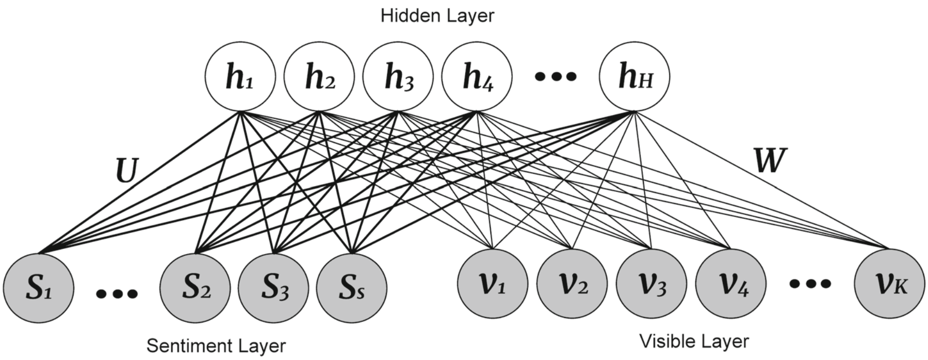


Fig. 3 Generative probabilistic proposed sentiment/topic model

data. As shown in Fig. 3, like RBM, this method has a two-layer structure, but with a vector corresponding to the document label or the number of existing classes added to the visible layer of the structure. The input vector of this structure in the visible part is a constant-length vector of the same size as the dictionary, where the number of word repetitions is specified.

As shown in Fig. 3, for each text document, the proposed model receives a binary vector representing the sentiment of document as input. The existing distributions over words in each topic and their associated sentiments are extracted in the hidden layer. In the presence of the additional layer and its associated parameters, the energy calculation equation is turned into:

$$E(\mathbf{V}, \mathbf{s}, \mathbf{h}) = -\sum_{j=1}^{H}\sum_{k=1}^{K} W_{kj} h_j \hat{v}_k - \sum_{j=1}^{H}\sum_{l=1}^{S} U_{lj} h_j s_l - \sum_{k=1}^{K} v_k a_k - \sum_{l=1}^{S} s_l c_l - D\sum_{j=1}^{H} h_j b_j \quad (6)$$

In (6), $\theta = \{W, U, \mathbf{a}, \mathbf{b}, \mathbf{c}\}$ is the set of model parameters where $W_{K\times H}$ is the weight matrix for the connection between the visible layer and the hidden layer, $U_{S\times H}$ is the weight matrix for the connection between the sentiment layer and the hidden layer, and **a**, **b** and **c** are the bias vectors of the visible, hidden, and sentiment layers, respectively. K and H are the sizes of the dictionary and the hidden layer, and S is defined as the number of existing sentiments or the size of the sentiment vector. The probability that the model assigns to each document and its associated sentiment layer is calculated as follows:

$$p(\mathbf{v}, \mathbf{s}, \mathbf{h}) = \frac{1}{Z} e^{-E(\mathbf{v},\mathbf{s},\mathbf{h})} \Rightarrow p(\mathbf{v}, \mathbf{s}) = \frac{1}{Z}\sum_{h} e^{-E(\mathbf{v},\mathbf{s},\mathbf{h})} \;;\; Z = \sum_{\mathbf{v}}\sum_{\mathbf{s}}\sum_{\mathbf{h}} e^{-E(\mathbf{v},\mathbf{s},\mathbf{h})} \quad (7)$$

In (7), $Z(\theta)$ is the partition function and ensures that the value obtained for configuration $(\mathbf{v}, \mathbf{s}, \mathbf{h})$ in (7) is an integer between 0 and 1.

In the proposed model, the values of visible, sentiment and hidden layers are calculated as follows:

$$p(v_i = w|\mathbf{h}) = \frac{exp(a_w + \sum_{j=1}^{H} W_{wj} h_j)}{\sum_{k=1}^{K} exp(a_w + \sum_{j=1}^{H} W_{wj} h_j)}; \quad (8)$$

$$p(s_l = 1|\mathbf{h}) = \frac{exp(c_l + \sum_{j=1}^{H} U_{lj} h_j)}{\sum_{l=1}^{S} exp(c_l + \sum_{j=1}^{H} U_{lj} h_j)}; \quad (9)$$

$$p(h_j = 1|\mathbf{v}, \mathbf{s}) = \sigma\left(Db_j + \sum_{k=1}^{K} W_{kj}\hat{v}_k + \sum_{l=1}^{S} U_{lj} s_l\right); \quad (10)$$

where σ is the logistic function.

The value of the hidden layer depends on the values of both visible and sentiment layers. Therefore, in (10), the value of the hidden layer is obtained by sampling a conditional distribution depending on the values of both visible and sentiment layers. The reason for the use of the softmax function for the visible and sentiment layers in both (8) and (9) is that once calculated the values of these layers conditioned to the hidden layer, need to be sampled related to these values. The use of the softmax function ensures that the values calculated for these two vectors are a polynomial probability distribution that can be easily sampled.

3.1 Training of the proposed model

The CD algorithm [4–6, 23] is used to train the proposed model and update the network parameters, including the weight matrices for the connections between the visible and hidden layers and between the sentiment and hidden layers, as well as the biases of all three layers. The model parameters are updated with the following equation:

$$\triangle\theta = \alpha \left(E_{P_{data}}[\theta] - E_{P_{model}}[\theta]\right) \Rightarrow \theta_{t+1} = \theta_t + \triangle\theta. \tag{11}$$

In (11), $E_{p_{data}}[.]$ is the expected value of the model parameters according to the data distribution and $E_{p_{model}}[.]$ is the expected value of the model parameters according to the distribution obtained by the model.

4 Empirical results

In this section, we explain the procedure of model testing and evaluation, and report and analyze the results obtained from different tests. The purpose of these tests is to observe the effect of adding a sentiment layer on topic modeling, sentiment tagging, classification, and information retrieval. In Sections 4.5 and 4.6, the proposed approach is compared with the RS model.

4.1 Description of datasets

Our tests and evaluations were performed by the use of several standard datasets of the field of topic modeling and sentiment analysis. A brief description of these databases is provided in the following.

The 20-Newsgroups (20NG) [10] dataset is a well-known dataset in the field of topic modeling. This dataset consists of 18,786 text documents collected from Usenet newsgroup repositories. This document collection is divided into 20 newsgroups, each related to one specific topic. Of the 18,786 documents in this dataset, we use 11,284 documents for the training and use 7,502 documents for testing the trained model. The 2000 most frequently repeated words in this dataset are used to compile the dictionary.

The movie review (MR) [18] dataset is another standard dataset for performance evaluation in the field of topic modeling. Our tests are conducted using the second version of this dataset, which includes 1,000 positive and 1,000 negative movie reviews collected from the Internet Movie Database (IMDB) website. The average length of each review in this dataset is 30 sentences.

The third dataset is the multi-domain sentiment (MDS) dataset introduced by Blitzer et al. in 2007 [3], which consists of collected reviews about four types of Amazon products:

Table 1 Movie Review Dataset Statistical Information. Avg is the mean document length, St. Dev. is the standard deviation in document length

Dataset	Dict size	# of Train docs	# of Test docs	Avg	St. Dev.
MR1	2000	1000	1000	90.18	40.23
MR2	10000	1000	1000	186.35	81.33
MR3	24916	1000	1000	299.75	126.51

Books, DVDs, electronics, and kitchen appliances. The MDS dataset contains 1,000 positive and 1,000 negative reviews for each of the above mentioned product types.

4.2 Preparation of datasets

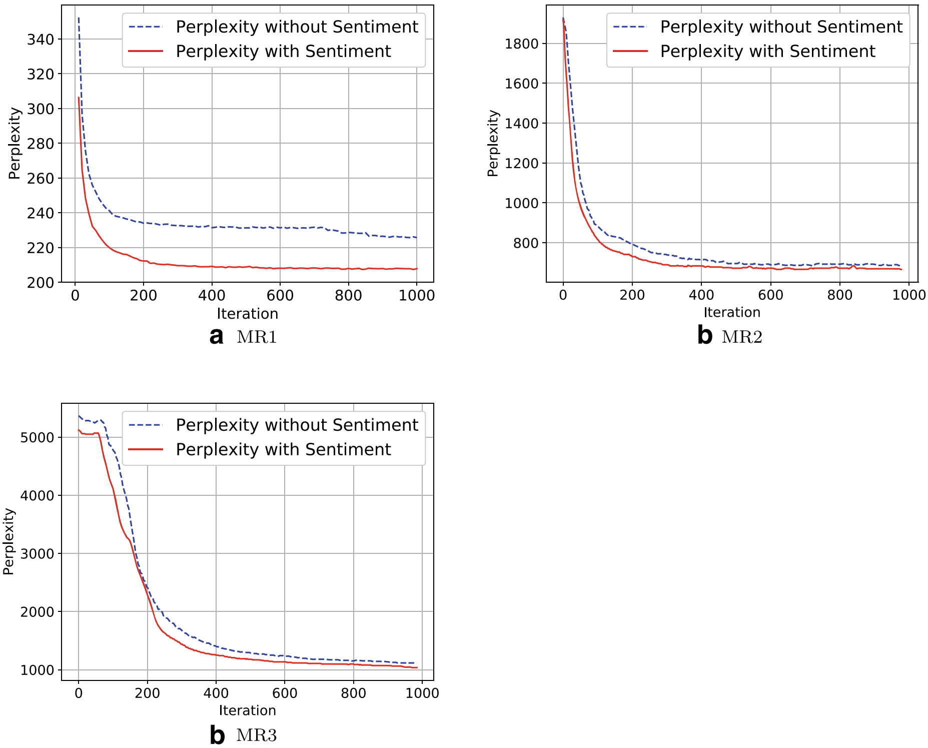
After preprocessing the texts of the MR dataset (removing stop words, stemming and lemmatizing), each document was converted into a sequence of words. In addition to the dictionary compiled from preprocessed data (with a size of 24,916 words), we also used 2000-word and 10000-word dictionaries belonging to the 20NG [10] and Reuters Corpus Volume I (RCV1) [13] datasets, respectively. We called these three states of Movie Review dataset MR1, MR2 and MR3. The statistics obtained from the described procedures are presented in Table 1. In the next step, we partitioned the database into two subsets, one for training and another for testing. Each of these subsets consisted of 1000 documents, 500 with positive tag and 500 with negative tag.

4.3 Sentiment lexicon

The sentiment lexicon is a pre-made general dictionary where for each word there are three sentiment tags, positive, negative, and neutral, each assigned with a weight between 0 and 1 so that the sum of all weights is 1. In this paper, we use a sentiment dictionary called MPQA [8]. This sentiment dictionary contains 4053 words, each with a 3-element vector, where the first element represents the neutrality weight, the second element represents the positivity weight, and the third element represents the negativity weight of the corresponding word. Overall, this dictionary contains 1511 positive words and 2542 negative words.

4.4 Details of training

We used the MR database as the input data to train the model in three different modes (MR1, MR2 and MR3) before testing its performance. The results of the conducted tests are presented later in the paper. The training on the all three states was performed using the first order CD algorithm. In all three training modes, the model was trained for 1000 iterations on the entire training subset with batch size of 1. The other parameter involving the training is the number of units in the hidden layers (h), which equals the number of topics. In all training modes, we trained the proposed method and the RS model for $h = \{5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 60, 70, 80, 90\}$. For all training modes, we used a learning coefficient $\alpha = 0.001$. The parameters W and U, which represent the weights of connections between the visible and hidden layers and between the sentiment and hidden layers, and the parameters $\mathbf{a}$ and $\mathbf{c}$, which are the biases of the visible and sentiment layers, were initialized with random numbers from a Gaussian distribution with a mean of 0 and variance 1. The bias of the hidden layer $\mathbf{b}$ was initialized at zero.



a MR1

b MR2

b MR3

Fig. 4 Evaluation of perplexity variation during the training on MR dataset

4.5 Document modeling and evaluation as a generative model

In this section, we present the results of performance evaluation of the proposed method as a generative probabilistic model in comparison with the RS model. As mentioned, these evaluations were performed after training with three dictionaries and using the documents in the test set. Through the analysis of the results, we show that the proposed method outperforms the RS model in probability estimation for the unobserved documents.

We use a criterion called perplexity to evaluate the calculated probability for the documents. Perplexity is a commonly used criterion for comparison of different probabilistic models in the NLP field. Perplexity has been defined as:

$$Perplexity = exp\left(-\frac{\sum_{n=1}^{N} \log p(\mathbf{v}_n)}{\sum_{n=1}^{N} D_n}\right). \tag{12}$$

According to the (12), perplexity equals the inverse of the mean per word likelihood obtained for each document on log-scale. In the modeling with an appropriate probabilistic model, perplexity should be monotonically decreasing. Overall, the lower is the model perplexity on the dataset, the better is the model quality.

Figure 4 shows the variation of perplexity of the proposed model and the RS model during the training with the MR dataset. As can be seen, in all three charts, the proposed joint sentiment/topic model has a greater perplexity decline than the RS topic model. Also, in all three charts, perplexity decline is sharper at the beginning of the training than at the final

Table 2 Perplexity estimation on movie review dataset using proposed model

TestSet type	Ppl without sentiment	Ppl with sentiment
MR1	423.89	**406.74**
MR2	2028.69	**1871.57**
MR3	5842.39	**5824.97**

stages. It can be observed from the 200th iteration onward, there is no significant change in perplexity. Careful examination of Fig. 4 reveals that adding the sentiment layer to construct a generative probabilistic model, as we did in this paper, leads to greater perplexity reduction in the training phase and therefore to the development of a better probabilistic method for document modeling.

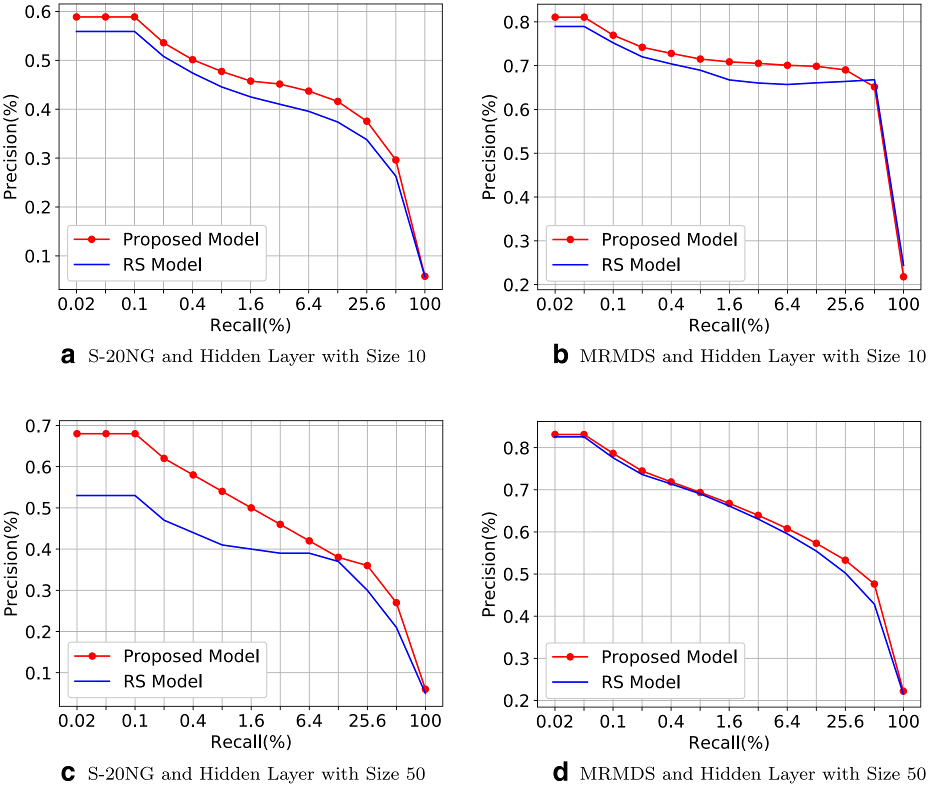
The calculated perplexity values presented in Table 2 are also the proof of higher performance of the proposed generative approach in the modeling process. The perplexity values shown in Table 2 are for the test set of the MR dataset and 2000-word, 10000-word, and 24916-word dictionaries. As shown in Table 2, the perplexity values obtained for the proposed model are lower than those obtained for the RS model. Thus, as stated earlier, the use of an additional layer dedicated to sentiment leads to development of a probabilistic document modeling method capable of outperforming the RS method.

4.6 Information retrieval

Since the proposed approach is a generative method for simultaneous modeling of topics and sentiments, the first requirement for the evaluation of this model in the data retrieval context is to use a dataset with both sentiment and topic labels for every document. Given the absence of such dataset, we created two datasets with both sentiment and topic tags for the testing purpose.

The first sentiment/topic database was created by assigning sentiment tags to the 20NG dataset. To do so, for each document we counted the number of words with known sentiment polarity using the MPQA sentiment dictionary. Then, the documents for which the number of positive words was greater than negative were given a positive tag and vice versa. We called this dataset Sentiment-20NG.

The second database created for the evaluation of the proposed method in the information retrieval context was created by compilation of the MR and MDS datasets introduced in Section 4.1. All of these 5 datasets (MDS alone consists of 4 different parts, each containing 2000 documents) only have sentiment labels. But each of these documents can be considered to represent a specific topic. Thus, these datasets were combined together to create a new larger dataset called MRMDS, which consists of 10000 documents, 5000 with positive tags and 5000 with negative tags, and five topics including: movie, book, DVD, electronic, and kitchen appliances. After the preprocessing phase, each of these documents was converted to the lib-svm file using the 2000-word dictionary of the 20NG dataset. Of the 10000 document obtained by combining these 5 datasets, 7500 documents with even sentiment label distribution (3750 documents labels with positive and 3750 documents with negative labels) and even topic distribution (1500 documents -consisting of 750 positive and 750 negative documents- per topic,) were assigned to the training set. The remaining 2500 documents (500 documents -consisting of 250 positive and 250 negative documents- per topic) were assigned to the testing set.



a S-20NG and Hidden Layer with Size 10

b MRMDS and Hidden Layer with Size 10

c S-20NG and Hidden Layer with Size 50

d MRMDS and Hidden Layer with Size 50

Fig. 5 Information Retrieval on S-20NG and MRMDS Datasets by Using Proposed Model and RS Model

The purpose of the evaluation is to assess the effects of considering sentiment on the information retrieval with the proposed model. For this evaluation, we used the precision-recall plot. This plot is a well-known criterion for the evaluation and comparison of information retrieval methods. To obtain this plot, the precision and recall values achieved by each model must be plotted against each other.

Figure 5 shows the results obtained by the evaluation of data retrieval performance of the proposed approach and the RS model. As can be seen in both charts of Fig. 5, and especially in Chart Fig. 5c, the proposed method has a better data retrieval performance than the RS model. The precision and recall values plotted in Fig. 5 were obtained as described in the following. First, we trained the proposed model only with sentiment labels (without topic labels) for 500 iterations using the hidden layers of size 10 and 50 units. For the RS model, training was performed without any label for 500 iterations using the hidden layers of size 10 and 50 units. Then, for each document in each testing set, we calculated the cosine similarity of the document with all documents of the training set to obtain precision and recall rates. Finally, the precision values obtained for the entire testing set were averaged and the charts of Fig. 5 were plotted.

In addition to the precision versus recall evaluation, there is another test (f-score evaluation) which is highly recommended in order to evaluate models in the information retrieval task. Accordingly, we performed a f-score evaluation for the proposed model in comparison

Table 3 Max-fscore evaluation of the proposed model and RS model

	Hidden neurons	RS model	Proposed model
MRMDS	10	**0.579**	0.573
S-20NG	10	0.347	**0.375**
MRMDS	50	0.466	**0.493**
S-20NG	50	0.297	**0.353**

to the RS model by using the values that we calculated for precision versus recall evaluation in Fig. 5. Actually, we used the following equation:

$$F - score = 2 * \frac{recall * precision}{recall + precision}; \tag{13}$$

to calculate different values for the f-score based on the correspondent values for recall and precision. After calculation of the f-score values we observed that the best value for f-score (max-fscore) for each configuration (different datasets and different number of hidden neurons). Indeed, we know the the best and maximum value for the f-score is 1, and it happens when the value of precision and recall are equal to 1; and the closer value to 1 for the f-score, the better performance in the information retrieval task. As we can observe from the Table 3 the proposed model has better max-fscore for three different settings in comparison to the RS model, and also for the MRMDS dataset with the size of hidden layer equal to 10 the result is competitive. Consequently, max-fscore evaluation confirms the idea of the proposed model which is considering and adding sentiment to the information retrieval task leads to a better performance.

4.7 Topic visualization

This section presents the results obtained by the use of MPQA sentiment dictionary to evaluate the precision of topic models in terms of sentiment label assignment. This evaluation is inspired by the test conducted elsewhere on well-known topic models such as DocNADE [11] and LDA [2].

Given the structure of the proposed approach (described in Section 3), we know that each hidden layer unit is connected to all units both in the visible layer and in the sentiment layer. Each unit in the sentiment layer is equivalent to a sentiment tag, and each unit in the visible layer corresponds to a word. In the topic modeling of text documents, each topic is defined as a polynomial probability distribution on all dictionary words, so we know that each unit of the hidden layer is connected, with a specific weight, to all dictionary words in the visible layer. For each word, this weight represents the significance of that word in that topic.

For this evaluation, we first calculated the total number of words shared between each of the three dictionaries and the MPQA sentiment lexicon. Table 4 shows the results obtained from this operation. Then, we followed the below procedure for each of the modeling modes and topic numbers:

1. Calculating the total weights of positive words and negative words for each topic by the use of the sentiment dictionary and the matrix weight for the connection between the visible and hidden layers.
2. Calculating, for each topic, the difference between the two values calculated in step 1 and sorting the answers in descending order.

Table 4 Number of words shared between each of the three dictionaries and the MPQA sentiment lexicon

	Total number	# Positive words	# Negative words
NG(2000)	155	100	55
RCV(10000)	950	447	503
MR(24916)	3114	1242	1872

3. Assigning positive tags to the top five topics of the ordered list (most positive topics); and assigning negative tags to the bottom five topics of this list (most negative topics).
4. Comparing the tags assigned to each topic with the corresponding topic weights in the connection of the sentiment layer to calculate the precision.

The idea behind the comparison made in step 4 (comparison of the tag assigned to each topic with the corresponding weight in the sentiment layer) is that for a topic assigned with a positive tag in step 3, the weight corresponding to the positive sentiment tag for that topic in the sentiment layer should be greater than the negative weight for the same topic and vice versa. Figure 6 shows the results of this evaluation. This figure indicates that as the dictionary size increases, so does the model precision in the assignment of sentiment tags to the topics. A comparison of the values presented for different dictionaries in Table 4 with Fig. 6 reveals the cause of the relationship between the model precision and the dictionary size. As can be seen, as the dictionary size increases, so does the number of words shared between dictionary and the sentiment dictionary, and this leads to a greater differentiation of the positive and negative topics in the training process, which result in improved model precision in the training and in assigning sentiment labels to the topics.

4.8 Sentiment classification

This section presents the results of the sentiment classification performed for the MR dataset using the proposed approach. We use a basic word count-based method to evaluate the sentiment classification precision of the proposed method in different modes. We also use the sentiment classification results obtained from the support vector machine (SVM) and

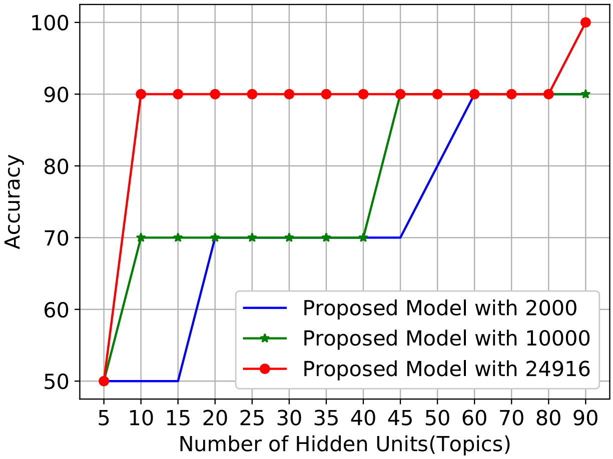
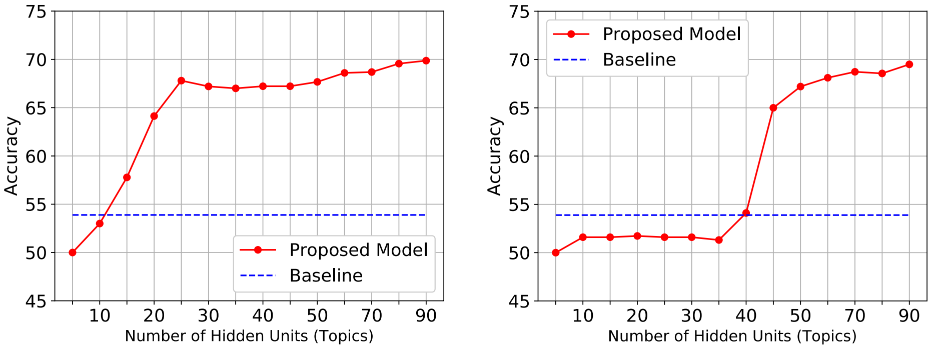


Fig. 6 Precision Evaluation on Sentiment Assignment to Topics



a Dictionary Size 2000

b Dictionary Size 10000

Fig. 7 Sentiment classification in movie review dataset with proposed model and base model for different number of topics

two two-layer neural networks, one initialized with random values and another initialized with the values given by the proposed method after training, to evaluate the parameters learned by the model.

Both neural networks used for comparison are of multilayer perceptron (MLP) type and utilize a cross-entropy error function. In both networks, the number of neurons in the first and second layers are equal to the number of topics and sentiments respectively. Also, the first and second layers of both networks operate based on the tanh activation function and the softmax function, respectively.

To calculate the precision of the basic model, we counted the words with a specific sentiment polarity in each document of the test set. In other words, for each document, we calculated the number of positive words and negative words using the MPQA sentiment lexicon. After listing the number of positive and negative words for each document, we assigned a positive label to any document for which the positive word count was greater than the negative, and assigned a negative label to the documents with the opposite property.

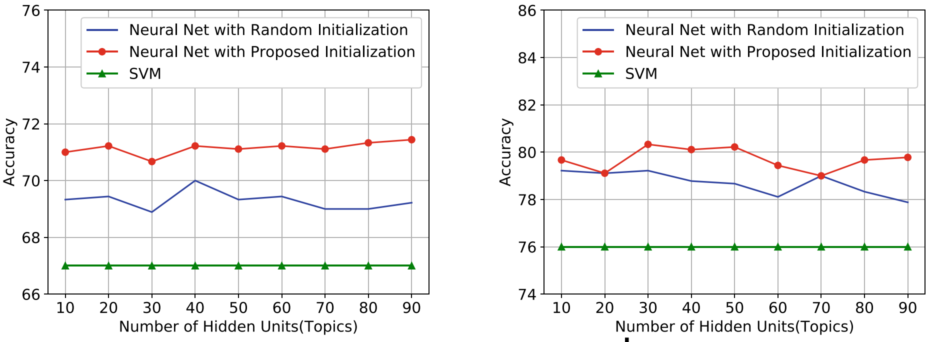
For sentiment classification using the proposed approach, we first used the following equation:

$$p(h_j = 1|\mathbf{V}) = \sigma\left(Db_j + \sum_{k=1}^{K} W_{kj}\hat{v}_k\right) \tag{14}$$

to obtain for each text document, the probability value of each hidden unit. The next step was to calculate the sentiment layer corresponding to the current document using (9). Since the value of this layer is given by a softmax function, it is in the form of a probability distribution where entries add up to 1. Then, for each document, we checked the values obtained for the sentiment layer, and assigned the document with the sentiment label corresponding to the greatest value observed in that layer.

The sentiment classification results obtained by the proposed approach and the basic model for two different datasets are presented in Fig. 7. To calculate the sentiment classification precision of the proposed model, we used the model trained for 1000 iterations with each dataset and different numbers of topics.

According to Fig. 7a and b, in both states, as the number of topics increases, so does the classification precision of the proposed model, and the extent to which it outperforms the basic model.



a Dictionary Size 2000 **b** Dictionary Size 10000

Fig. 8 Sentiment classification in movie review dataset

Figure 8 shows the precision of the sentiment classification performed by two two-layer Neural Networks and SVM classification. We used the parameters (weight matrix and bias) obtained in 1000 iterations of training of the proposed model to initialize one of the neural networks and the other one was initialized at random. According to Fig. 8, in both dataset states, the neural network initialized with the values learned by the proposed method has a better precision than the other two methods.

As shown in Fig. 8b, when using the 10000-word dictionary, the two networks have the same precision only when the number of topics is either 20 or 70. In other cases, the network with non-random initialization has outperformed all other models. In general, we can conclude that the neural network initialized with the values learned by the proposed method has a better sentiment classification performance than the other two models.

5 Conclusion

In this paper, we presented a novel neural network-based model for the joint sentiment/topic modeling of text data. A review of literature showed the presence of only two Bayesian models, ASUM and JST, for this joint sentiment/topic modeling, and the features and limitations of these model were briefly discussed. The recent developments in the qualities and use of neural networks and the absence of any neural network-based method in the field of joint sentiment/topic modeling were the factors that encouraged the authors to try this approach for this application.

We proposed a supervised neural network-based approach for the joint sentiment/topic modeling of text data. The proposed approach, which falls in the category of generative probabilistic methods, is an extension of the RS model based on the Restricted Boltzmann Machine (RBM) neural network. In the proposed approach, the model is equipped with an additional layer of polynomial probability distribution nature to enable the hidden layer to learn better and more distinct features for each document. This model was trained using a gradient approximation method known as the Contrastive Divergence algorithm.

The proposed model was evaluated using the movie review dataset, the 20-newsgroups dataset, and the multi-domain sentiment dataset, which are the prominent databases for the performance evaluation of topic and sentiment models of text data. We also used perplexity, which is a well-known criterion for the evaluation of generative models, to evaluate the performance of the proposed method in the text data modeling. According to the results, we can claim that incorporating the sentiment into the document modeling, as we did in the present work, will lead to the development of generative models of higher quality for document modeling. We also evaluated the data retrieval performance of the proposed method through comparison with the RS model. The results of the tests performed on two databases demonstrated the superior performance and precision of the proposed method in data retrieval from text documents.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

1. Blei DM (2012) Probabilistic topic models. Commun ACM 55(4):77–84. https://doi.org/10.1145/2133806.2133826
2. Blei DM, Ng AY, Jordan MI (2003) Latent dirichlet allocation. J Mach Learn Res 3(Jan):993–1022
3. Blitzer J, Dredze M, Pereira F et al (2007) Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. ACL 7:440–447
4. Carreira-Perpinan MA, Hinton G (2005) On contrastive divergence learning. 33–40
5. Hinton G (2010) A practical guide to training restricted boltzmann machines. Momentum 9(1):926
6. Hinton GE (2002) Training products of experts by minimizing contrastive divergence. Neural Comput 14(8):1771–1800
7. Hinton GE, Salakhutdinov RR (2009) Replicated softmax: an undirected topic model. In: Advances in neural information processing systems, pp. 1607–1614
8. Jain TI, Nemade D (2010) Recognizing contextual polarity in phrase-level sentiment analysis. Int J Comput Appl IJCA 7(5):5–11
9. Jo Y, Oh AH (2011) Aspect and sentiment unification model for online review analysis. pp 815–824. ACM
10. Lang K (1995) Newsweeder: Learning to filter netnews. In: Proceedings of the 12th international conference on machine learning, pp 331–339
11. Larochelle H, Lauly S (2012) A neural autoregressive topic model. pp 2708–2716
12. Larochelle H, Murray I (2011) The neural autoregressive distribution estimator. 29–37
13. Lewis DD, Yang Y, Rose TG, Li F (2004) Rcv1: A new benchmark collection for text categorization research. J Mach Learn Res 5(Apr):361–397
14. Li Q, Yang Y (2016) Topic correlation model for cross-modal multimedia information retrieval. Pattern Anal Appl 19:1007–1022
15. Lin C, He Y, Everson R, Ruger S (2012) Weakly supervised joint sentiment-topic detection from text. IEEE Trans Knowl Data Eng 24(6):1134–1145
16. Lyang T, Zhiwei N Emerging opinion leaders in crowd unfollow crisis: a case study of mobile brands in twitter. Pattern Analysis and Application
17. Mohr JW, Bogdanov P (2013) Introduction—topic models: What they are and why they matter. Poetics 41(6):545–569
18. Pang B, Lee L (2004) A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In: Proceedings of the ACL, polarity dataset v2.0. http://www.cs.cornell.edu/people/pabo/movie-review-data/. Accessed: 2017-04
19. Pang B, Lee L, Vaithyanathan S (2002) Thumbs up?: sentiment classification using machine learning techniques. In: Proceedings of the ACL-02 conference on empirical methods in natural language processing-volume 10, pp 79–86. Association for Computational Linguistics
20. Pang B, Lee L et al (2008) Opinion mining and sentiment analysis. Found Trends Inf Retr 2:1–135
21. Smolensky P (1986) Parallel distributed processing: Explorations in the microstructure of cognition. In: Information processing in dynamical systems: Foundations of harmony theory, vol 1. MIT Press, Cambridge, pp 194–281. http://dl.acm.org/citation.cfm?id=104279.104290
22. Steyvers M, Griffiths T (2007) Probabilistic topic models. Handbook of latent semantic analysis 427(7):424–440
23. Woodford O (2013) Notes on contrastive divergence. Department of Engineering Science. University of Oxford, Tech. Rep, Oxford

Masoud Fatemi recieved the B.S. degree from Shahrood University of technology, Shahrood, Iran in 2014 and the M.S. degree from Isfahan University of Technology, Isfahan, Iran in 2017. His current interests include machine learning, pattern recognition and data mining.

Mehran Safayani received his B.S. degree in computer engineering from Isfahan University, Isfahan, Iran in 2002. Then, he received the M.Sc. and Ph.D. degrees from Sharif University of Technology, Tehran, Iran in computer architecture and artificial intelligence in 2006 and 2011 respectively. Since 2012, he is an assistant professor of Electrical and Computer Engineering at Isfahan University of Technology. His research interests include machine learning, neural networks and deep learning, statistical pattern recognition and soft computing.