Siamese Network-Based Supervised Topic Modeling

Minghui Huang[1,*], **Yanghui Rao**[1,†], **Yuwei Liu**[1], **Haoran Xie**[2], **Fu Lee Wang**[3]
[1]School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China
[2]Department of Mathematics and Information Technology,
The Education University of Hong Kong, Tai Po, Hong Kong
[3]School of Science and Technology,
The Open University of Hong Kong, Ho Man Tin, Kowloon, Hong Kong
huangmh25@mail2.sysu.edu.cn, raoyangh@mail.sysu.edu.cn,
liuyw23@mail2.sysu.edu.cn, hrxie2@gmail.com,
pwang@ouhk.edu.hk

Abstract

Label-specific topics can be widely used for supporting personality psychology, aspect-level sentiment analysis, and cross-domain sentiment classification. To generate label-specific topics, several supervised topic models which adopt likelihood-driven objective functions have been proposed. However, it is hard for them to get a precise estimation on both topic discovery and supervised learning. In this study, we propose a supervised topic model based on the Siamese network, which can trade off label-specific word distributions with document-specific label distributions in a uniform framework. Experiments on real-world datasets validate that our model performs competitive in topic discovery quantitatively and qualitatively. Furthermore, the proposed model can effectively predict categorical or real-valued labels for new documents by generating word embeddings from a label-specific topical space.

1 Introduction

As one of the most widely used text mining techniques, topic modeling can extract meaningful descriptions (i.e., topics) from a corpus (Blei, 2012). Most previous topic models, such as probabilistic Latent Semantic Analysis (pLSA) (Hofmann, 1999) and Latent Dirichlet Allocation (LDA) (Blei et al., 2001) are unsupervised. In unsupervised topic models, each document is defined as a mixture distribution over topics and each topic is represented as a mixture distribution over words. Unsupervised topic models only exploit words in documents and do not incorporate the guidance of labels into learning processes. Therefore, these models fail to discover label-specific topics, which are important to support personality psychology (Weiner and Graham, 1990), aspect-level sentiment analysis (Liu, 2012), and cross-domain sentiment classification (He et al., 2011). For example, label-specific topics generated from sentimental texts can help to find attributions and causes for different sentiments by associating sentiments with real-world topics/events.

In light of this consideration, several supervised topic models are proposed to generate label-specific topics. One of the most representative models is the supervised Latent Dirichlet Allocation (sLDA) (Blei and McAuliffe, 2007), which restricts a document being associated with one real-valued response variable. To deal with categorical labels, multi-class sLDA (sLDAc) (Wang et al., 2009) and Labeled Latent Dirichlet Allocation (L-LDA) (Ramage et al., 2009) are proposed, but they are only applicable to classification. Recently, a supervised Neural Topic Model (sNTM) (Cao et al., 2015) is developed to tackle supervised tasks of both classification and regression. As a hybrid method, sNTM is in essence a neural network by following the document-topic distribution in topic models. Unfortunately, the label information has a little effect on topic generation since sNTM models documents and labels separately.

The above limitation motivates us to develop a supervised topic model which can jointly model documents and labels. Particularly, we propose a Siamese Labeled Topic Model (SLTM) to exploit the information of documents and labels based on the Siamese network (Bromley et al., 1993; Hu et al., 2014; Wang and Zhang, 2017), where weight matrices in SLTM represent conditional distributions. Therefore, by constraining weight matrices during the learning procedure, SLTM can follow probabilistic characteristics of topic models strictly. Compared to previous supervised topic models, the main advantages of our SLTM are summarized as follows. First, SLTM can gener-

[*] This work was finished when the first author was an undergraduate student of her final year.

[†] The corresponding author.

Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 4652–4662
Brussels, Belgium, October 31 - November 4, 2018. ©2018 Association for Computational Linguistics

ate more coherent label-specific topics than others. This is because the supervision of labels is incorporated into topic modeling for SLTM. On the other hand, the mapping of topics to labels is unconstrained for most existing supervised topic models, which renders many coherent topics being generated outside labels. Second, strengths of neural networks are incorporated into SLTM to bootstrap its inference power on label prediction. Third, each word can be mapped to a topical embedding space and represented by a word embedding after generating label-specific topics.

To validate the effectiveness of the proposed model, we evaluate it on two real-world datasets in text mining. Experimental results indicate that our method is able to discover more coherent and label-specific topics than baseline models. Moreover, word embeddings learned by the proposed model can be used to predict labels for new documents effectively.

The remainder of this paper is organized as follows. We summarize related studies on supervised topic modeling in Section 2. For convenience of describing our model, we present the neural network view of topic models in Section 3. Then, we detail the proposed SLTM in Section 4. Experimental design and analysis of results are shown in Section 5. Finally, we present conclusions and future work in Section 6.

2 Related Work

Topic models, which focus on discovering unobserved class variables named "topics" statistically, have been widely used in text mining. One of the early topic models is pLSA (Hofmann, 1999). In pLSA, a document's word vector was decomposed into a mixture of topics, and a topic was represented as a probability distribution over words. LDA (Blei et al., 2001) extended pLSA by adding Dirichlet priors for a document's multinomial distribution over topics and a topic's multinomial distribution over words, which makes it suitable to generate topics for unseen documents.

The aforementioned models are unsupervised, which may be computationally costly to do some task-specific transformation when there is extra labeling information (Cao et al., 2015). To address this issue, several supervised topic models have been proposed to introduce the label guidance in learning processes. One of the most widely used supervised topic models is sLDA (Blei and McAuliffe, 2007). In sLDA, each document was paired with a response variable which obeys the Gaussian distribution. By extending the sLDA, BP-sLDA (Chen et al., 2015) applied back propagation over a deep architecture in conjunction with stochastic gradient/mirror descent for model parameter estimation, leading to scalable and end-to-end discriminative learning characteristics. Based on sLDA, multi-class sLDA (sLDAc) (Wang et al., 2009) was proposed to model documents with categorical labels by adding a softmax classifier rather than a linear regression in sLDA to a standard LDA. Another method of tackling corpora with discrete labels is L-LDA (Ramage et al., 2009), which associated each label with only one topic. To improve the performance of L-LDA in the classification task, Dependency-LDA (Dep-LDA) (Rubin et al., 2012) incorporated an extra topic model to capture the dependencies between labels and took the label dependencies into consideration when estimating topic distributions. Recently, a nonparametric supervised topic model (Li et al., 2018) was proposed to predict the response of interest (e.g., product ratings and sales). The limitation of above models is that they are only applicable to either discrete or continuous data.

In this paper, we propose a Siamese network-based supervised topic model named SLTM. The most relevant work to SLTM is the supervised Neural Topic Model (sNTM) for both classification and regression tasks (Cao et al., 2015), which constructed two hidden layers to generate the n-gram topic and document-topic representations. However, different from our SLTM using bag-of-words methods, sNTM adopted fixed embeddings trained on external resources (Mikolov et al., 2013). Thus, sNTM can not learn data-specific topics. Furthermore, sNTM is hard to follow probabilistic characteristics of the topic-word distribution in topic models, because a topic generated by sNTM is composed of an infinite number of n-grams. Finally, sNTM modeled documents and labels separately rather than uniformly in our SLTM.

3 Preliminaries

For convenience of describing the proposed model, we use hollow uppercase letters (e.g., $\mathbb{D}$) to represent collections, bold uppercase letters (e.g., $\mathbf{W}_1$) to represent matrices, bold lowercase letters (e.g., $\mathbf{y}_i$) to represent vectors, regular uppercase letters (e.g., M) to represent scalar constants,

Table 1: Frequently used notations.

Notation	Description
M	Number of documents
K	Number of topics
N	Size of the vocabulary
L	Size of the label set
$\mathbb{D}$	Document collection
$d_i \in \mathbb{D}$	The i-th document
$\mathbb{V}$	Vocabulary
$v_j \in \mathbb{V}$	The j-th word
$\mathbb{Z}$	Topic collection
$z_k \in \mathbb{Z}$	The k-th topic
$\mathbb{Y}$	Label collection
$\mathbf{y}_i \in \mathbb{R}^L$	Labels for document d_i
$p(v_j \vert d_i)$	The probability of v_j given d_i

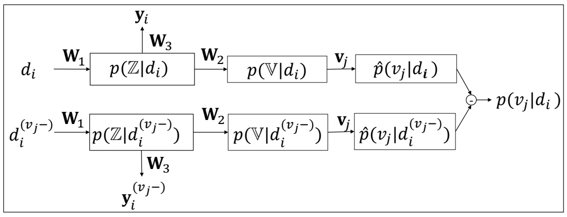

Figure 1: SLTM's word generation framework.

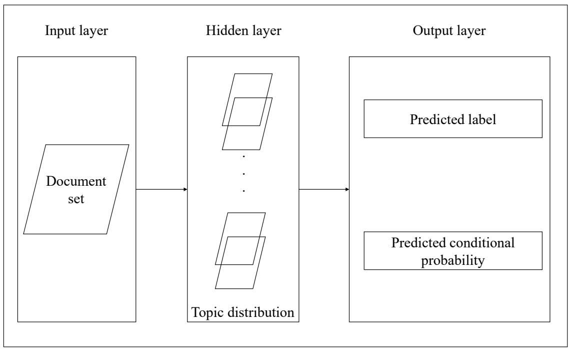

Figure 2: SLTM's architecture from the perspective of neural networks.

and regular lowercase letters (e.g., v_j) to represent scalar variables. Based on the above convention, frequently used notations are shown in Table 1. Given a document d_i with labels $\mathbf{y}_i$, our goal is to discover topics with a neural network framework. Therefore, we first describe the neural network view of topic models briefly.

Topic modeling is a popular latent variable inference method for co-occurrence data which associates unobserved classes with observations d_i and v_j, where v_j is a word in d_i. The conditional probability $p(v_j|d_i)$ is defined as:

$$p(v_j|d_i) = \sum_{k=1}^{K} p(v_j|z_k)p(z_k|d_i). \tag{1}$$

Let $\phi(v_j) = [p(v_j|z_1), \ldots, p(v_j|z_K)]$ and $\theta(d_i) = [p(z_1|d_i), \ldots, p(z_K|d_i)]$, then $p(v_j|d_i)$ in Equation 1 can be represented as the following vector form:

$$p(v_j|d_i) = \phi(v_j) \cdot \theta(d_i). \tag{2}$$

We represent horizontal stack by commas and vertical stack by semicolons, thus $\mathbf{W}_1 = [\theta(d_1)^T, \ldots, \theta(d_M)^T] \in \mathbb{R}^{K \times M}$ and $\mathbf{W}_2 = [\phi(v_1); \ldots; \phi(v_N)] \in \mathbb{R}^{N \times K}$, which are constrained by: $\mathbf{W}_1[k, m] \geq 0$, $\mathbf{W}_2[n, k] \geq 0$, $\sum_{k=1}^{K} \mathbf{W}_1[k, m] = 1$, and $\sum_{j=1}^{N} \mathbf{W}_2[j, k] = 1$, where $k \in [1, K]$, $m \in [1, M]$, and $n \in [1, N]$. Then, the vector form in Equation 2 can be extended to:

$$\begin{aligned} p(\mathbb{V}|\mathbb{D}) &= \begin{bmatrix} p(v_1|d_1) & \cdots & p(v_1|d_M) \\ \vdots & \ddots & \vdots \\ p(v_N|d_1) & \cdots & p(v_N|d_M) \end{bmatrix} \\ &= \begin{bmatrix} (\phi(v_1) \cdot \theta(d_1)) & \cdots & (\phi(v_1) \cdot \theta(d_M)) \\ \vdots & \ddots & \vdots \\ (\phi(v_N) \cdot \theta(d_1)) & \cdots & (\phi(v_N) \cdot \theta(d_M)) \end{bmatrix} \\ &= \mathbf{W}_2\mathbf{W}_1. \end{aligned} \tag{3}$$

With Equation 3, topic models can be viewed as neural networks, where $\mathbb{D}$ and $\mathbb{V}$ are input sets, $p(\mathbb{V}|\mathbb{D})$ is the output set, and $\mathbf{W}_1$ and $\mathbf{W}_2$ are parameter matrices of the neural network.

4 Siamese Labeled Topic Model

Similar to generative models such as pLSA, we propose a Siamese Labeled Topic Model (SLTM) based on the aforementioned neural network perspective of topic models. Figure 1 illustrates the framework of generating each word in SLTM, and the process is as follows. For a document d_i in $\mathbb{D}$, the topic distribution $p(\mathbb{Z}|d_i)$ is estimated by:

$$p(\mathbb{Z}|d_i) = \mathbf{W}_1\mathbf{d}_i, \tag{4}$$

where $\mathbf{d}_i$ is the indicator vector (Yang et al., 2013) of d_i, which means that the i-th entry of $\mathbf{d}_i$ is 1 and other entries are 0. Labels of d_i are $\mathbf{y}_i$, which are generated from the topic distribution of d_i as: $\mathbf{y}_i = \mathbf{W}_3 p(\mathbb{Z}|d_i)$. The above equation is constrained by $\mathbf{W}_3[l,k] \geq 0$, where $l \in [1,L]$, $k \in [1,K]$, and $\sum_{l=1}^{L} \mathbf{W}_3[l,k] = 1$ if $L > 1$. A topic z_k in $\mathbb{Z}$ has its word distribution $p(\mathbb{V}|z_k)$, which is computed by:

$$p(\mathbb{V}|z_k) = \mathbf{W}_2 \mathbf{z}_k, \tag{5}$$

where $\mathbf{z}_k$ is the indicator vector of z_k. Therefore, words can be generated from d_i as:

$$p(\mathbb{V}|d_i) = \mathbf{W}_2 \mathbf{W}_1 \mathbf{d}_i. \tag{6}$$

The architecture of SLTM from the perspective of neural networks is shown in Figure 2. With respect to the model optimization, we adopt the contrastive objective function used in previous works (Socher et al., 2014; Cui et al., 2014; Cao et al., 2015; He et al., 2017). For document d_i and every word v_j in d_i, we randomly sample a document from the document set $\mathbb{D}$ which does not contain v_j, as a negative sample document. The negative sample document is represented as $d_i^{(v_j-)}$ and has labels $\mathbf{y}_i^{(v_j-)}$. As shown in Figure 1, the lower sub-network, which takes $d_i^{(v_j-)}$ as input, has the same architecture as the the upper sub-network, which takes d_i as input. Because the document-topic distribution and the topic-word distribution of a corpus are fixed, $\mathbf{W}_1$, $\mathbf{W}_2$ and $\mathbf{W}_3$ are shared among two sub-networks of our model. These two sub-networks are twin networks and thus the proposed model is essentially the Siamese network. Our objective is to make word v_j be learned by topics in document d_i, while not be learned by topics in the negative sampled document $d_i^{(v_j-)}$. Therefore, we only take word v_j in $\mathbb{V}$ into consideration during the learning procedure, which can be implemented as dot-multiplying $p(\mathbb{V}|d_i)$ with the indicator vector of v_j (i.e., $\mathbf{v}_j$) as: $\hat{p}(v_j|d_i) = p(\mathbb{V}|d_i) \cdot \mathbf{v}_j$. Particularly, the objective is to make the predicted conditional probability $\hat{p}(v_j|d_i)$ approach the observed conditional probability $p(v_j|d_i)$ (i.e., term frequency of word v_j in document d_i), while make the conditional probability $\hat{p}(v_j|d_i^{(v_j-)})$ approach zero. Thus, the loss function of predicted conditional probabilities and the observed conditional probability can be defined as:

$$\begin{aligned} &loss(d_i, d_i^{(v_j-)}) \\ &\quad = |p(v_j|d_i) - \hat{p}(v_j|d_i) + \hat{p}(v_j|d_i^{(v_j-)})|. \end{aligned} \tag{7}$$

Algorithm 1 Training Algorithm for SLTM

```
Input: $\mathbb{S} = \{\mathbb{D}, \mathbb{Y}\}$;
repeat
    for all $(d_i, \mathbf{y}_i) \in \mathbb{S}$ do
        for each word $v_j$ in document $d_i$ do
            Sample a document $d_i^{(v_j-)}$ which does not contain $v_j$;
            Calculate $loss(SLTM)$;
            if $loss(SLTM)$ is reducing then
                Update $\mathbf{W}_1$, $\mathbf{W}_2$ and $\mathbf{W}_3$;
            end if
        end for
    end for
until convergence
```

We use another loss function $loss(\mathbf{y}_i, \mathbf{y}_i^{(v_j-)})$ to capture labels of d_i and $d_i^{(v_j-)}$, where $loss(\mathbf{y}_i, \mathbf{y}_i^{(v_j-)}) = loss(\mathbf{y}_i) + loss(\mathbf{y}_i^{(v_j-)})$. In the above, equations of $loss(\mathbf{y}_i)$ and $loss(\mathbf{y}_i^{(v_j-)})$ depend on the property of labels. For categorical and real-valued labels, the cross-entropy (Tang et al., 2014) and the mean absolute error (Willmott and Matsuura, 2005) are adopted, respectively.

The maximization of the weighted sum of conditional likelihoods is equivalent to minimize the losses of the weighted sum of loss functions, and these two loss functions are weighted by a hyper-parameter α as in (Tang et al., 2014). Thus, the loss function of SLTM is:

$$\begin{aligned} loss(SLTM) &= \alpha \times loss(d_i, d_i^{(v_j-)}) \\ &\quad + (1-\alpha) \times loss(\mathbf{y}_i, \mathbf{y}_i^{(v_j-)}). \end{aligned} \tag{8}$$

The effect of α on predicting labels and discovering topics will be investigated in Section 5.6. Based on $loss(SLTM)$, three kinds of weights, i.e., $\mathbf{W}_1$, $\mathbf{W}_2$, and $\mathbf{W}_3$ can be updated together by a vanilla back propagation (BP) algorithm with the early stopping criteria (Bengio, 2012). The training algorithm is shown in Algorithm 1.

After training, we obtain both document-topic and topic-word distributions. Then, each word can be mapped to a topic-level embedding space and represented as a word embedding. For instance, the word embedding of v_j is generated from the topic-word distribution $\mathbf{W}_2$ as:

$$\mathbf{e}(v_j) = \mathbf{W}_2[j, :]. \tag{9}$$

The generated word embeddings can be used for specific applications, such as label prediction. Particularly, we firstly represent a new document d_n by its document embeddings $\mathbf{e}(d_n)$, where $\mathbf{e}(d_n)$ is the sum of word embeddings of all words in d_n. Then, the predicted labels $\hat{\mathbf{y}}_n$ of document d_n can be estimated by:

$$\hat{\mathbf{y}}_n = f(\mathbf{W}_4 \mathbf{e}(d_n)), \tag{10}$$

where $\mathbf{W}_4$ denotes weights of each topic contributing to labels, and $f(.)$ is the activation function which depends on the type of labels. For categorical and normalized real-valued labels, we respectively adopt softmax and sigmoid as activation functions. Note that we do not predict labels for new documents based on $\mathbf{W}_3$ directly, because topic distributions of these documents can only be learned without the supervision of labels, i.e., new documents' topic distributions may be inconsistent to $\mathbf{W}_3$. Finally, we update $\mathbf{W}_4$ and word embeddings by RMSprop (Tieleman and Hinton, 2012) for label prediction.

5 Experiments

In this section, we firstly describe datasets and the setting of experiments. Secondly, we investigate the quality of generated topics by the topic coherence score and qualitative analysis. Thirdly, the quality of generated word embeddings is evaluated by label prediction and word similarity. Finally, the effect of the hyper-parameter α is evaluated on coherence of topics and label prediction.

5.1 Datasets and Setting

To evaluate the effectiveness of our method comprehensively, we conduct experiments on two real-world datasets with categorical and real-valued labels, respectively. The first corpus named ISEAR contains a collection of 7,666 sentences and each item is manually tagged with a categorical label over 7 emotions (Scherer and Wallbott, 1994). The second dataset YouTube[1] is often used for sentiment strength detection, which contains 3,407 comments on videos and each item is labeled with a real value between 0.1 (i.e., very negative sentiment) and 0.9 (i.e., very positive sentiment). These two datasets are selected for their similar word numbers in average. After removing stop words, the mean numbers of words in each document are 8.53 and 8.56 for ISEAR and YouTube. Besides, it is appropriate to evaluate the model performance on predicting emotions and sentiment strengths, because topics play an important role in understanding sentences or user comments (Liu, 2012). Since the proposed SLTM is suitable to both topic discovery and classification/regression tasks, we employ five kinds of baselines for comparison.

The first kind are the support vector machine (SVM), an efficient deep learning model for classification (i.e., fastText) (Grave et al., 2017), and the following supervised topic models which are confined to categorical labels:

- sLDAc (Wang et al., 2009): it models documents with categorical labels by adding a softmax classifier to a standard LDA.
- L-LDA (Ramage et al., 2009): it is a supervised model which associates labels with topics by one-to-one correspondence. Accordingly, the number of topics in L-LDA must equal the size of the label set.
- Dep-LDA (Rubin et al., 2012): it extends L-LDA by introducing a multinomial distribution over labels and capturing the dependencies between labels. Then, the label dependencies are used to sample topic distributions in supervised learning.

The second kind are the support vector regression (SVR), a state-of-the-art deep learning model for sentiment strength detection (i.e., HCNN) (Chen et al., 2017), and the following supervised topic models which are developed for predicting real-valued labels only:

- sLDA (Blei and McAuliffe, 2007): it is a classical supervised topic model, in which, each document is paired with a response variable, and the variable is defined as a Gaussian distribution with a mean value that is computed by a linear regression of topics.
- BP-sLDA (Chen et al., 2015): it applies back propagation over a deep architecture together with stochastic gradient/mirror descent for

[1] http://sentistrength.wlv.ac.uk/

Table 2: Topic coherence scores on ISEAR using different numbers of top words T.

	5	10	15
pLSA	0.0051	0.0024	-0.0013
LDA	0.0954	0.0492	0.0014
sLDAc	0.0014	0.0031	-0.0035
sNTM	-0.9267	-0.9508	-0.9667
SLTM	**0.1142**	**0.0680**	**0.0025**

Table 3: Topic coherence scores on YouTube using different numbers of top words T.

	5	10	15
pLSA	-0.0535	-0.2435	-0.3829
LDA	**-0.0154**	-0.2142	-0.3618
sLDA	-0.0627	-0.2502	-0.3962
BP-sLDA	-0.7021	-0.7670	-0.7900
sNTM	-0.9138	-0.9253	-0.9376
SLTM	-0.0993	**-0.1967**	**-0.3268**

parameter estimation of sLDA. The number of hidden layers is set to 3.

The third kind is a supervised n-gram model named sNTM, which is applicable to predict both categorical and real-valued labels for new documents (Cao et al., 2015). In sNTM, each n-gram is represented by a 300-dimensional embedding vector using the available tool word2vec[2]. By following (Cao et al., 2015), a large-scale Google News dataset with around 100 billion words is adopted for training. For topic discovery, two unsupervised topic models, pLSA (Hofmann, 1999) and LDA (Blei et al., 2001), are used as the fourth kind of baselines. Finally, we adopt two hybrid methods by combining LDA and supervised learning algorithms as baselines. In particular, a softmax classifier and a liner regression (LR) are used to predict categorical and real-valued labels for documents, respectively. Unless otherwise specified, we set α to 0.5 and adopt the stochastic gradient descent with batch size of 100 for training SLTM.

5.2 Coherence Score of Topics

Table 4: Each label's top 5 words on ISEAR.

Labels	Models	Top 5 words of label-specific topics
fear	sLDAc	home night car **afraid fear**
	L-LDA/Dep-LDA	night **afraid** car home **fear** felt
	SLTM	night **afraid fear** car home dark
joy	sLDAc	year passed heard exam university
	L-LDA/Dep-LDA	friend got time passed felt
	SLTM	**happy joy** passed got university
guilt	sLDAc	did didn't asked **guilty** said
	L-LDA/Dep-LDA	felt **guilty** friend did mother
	SLTM	**guilty** felt mother did friend
disgust	sLDAc	saw man **disgusted disgust** woman
	L-LDA/Dep-LDA	**disgusted** saw felt people friend
	SLTM	**disgusted** saw people man **disgust**
shame	sLDAc	know **ashamed** teacher happened lot
	L-LDA/Dep-LDA	**ashamed** felt friend time did
	SLTM	**ashamed** felt **shame** class teacher
anger	sLDAc	**angry** called new **anger** expected
	L-LDA/Dep-LDA	friend **angry** did time told
	SLTM	**angry** friend **anger** brother told
sadness	sLDAc	father close died away years
	L-LDA/Dep-LDA	died friend **sad** felt time
	SLTM	died **sad** death away friend

To investigate the quality of topics discovered by SLTM quantitatively, we use the topic coherence score based on the normalised pointwise mutual information (Lau et al., 2014) as the evaluation metric. Intuitively, a topic coherence score that is larger indicates that the quality of topics is better. All unsupervised topic models (i.e., pLSA and LDA) and supervised methods which associate one label with multiple topics (i.e., sLDAc, sLDA, BP-sLDA, and sNTM) are adopted for comparison. Although L-LDA and Dep-LDA can identify label-specific topics on ISEAR, these models' one-to-one mapping of labels and topics makes them unsuitable in this evaluation. Particularly, L-LDA and Dep-LDA constraint each topic to words in certain documents with the same label, which renders their coherence scores being estimated by a subset of the corpus only. On the other hand, the quality of topics is evaluated on the whole corpus for SLTM and other baseline models.

The average coherence scores of topics generated by different models on ISEAR and YouTube are respectively shown in Table 2 and Table 3, where the number of topics is 20, the number of top words T is set to 5, 10, and 15, and the best scores are highlighted in boldface. The results indicate that SLTM can discover more coherent topics than both unsupervised topic models and supervised methods, except for $T = 5$ on YouTube. It is also interesting to observe that supervised baseline models (i.e., sLDAc, sLDA, BP-sLDA, and sNTM) perform worse than pLSA and LDA for most cases, which validates that it is challenging to trade off label-specific word distributions with document-specific label distributions (Ramage et al., 2009).

5.3 Qualitative Analysis on Topics

In this part, we conduct qualitative inspection of 20 topics generated by SLTM. The ISEAR dataset

[2] https://code.google.com/p/word2vec/

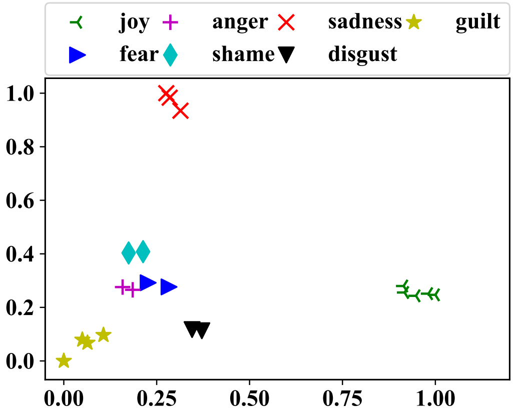

Figure 3: Scatter plot of topics identified by SLTM on ISEAR, where each point indicates a topic.

Table 5: Classification performance on ISEAR.

	Accuracy	Cohen's kappa
SVM	0.5063	0.4240
fastText	0.5104	0.4298
LDA+softmax	0.1506	0.0089
sLDAc	0.1875	0.0540
L-LDA	0.4650	0.3758
Dep-LDA	0.4888	0.4036
sNTM	0.2478	0.1212
SLTM	**0.5213**	**0.4415**

which contains multiple labels is used for illustration, since it is inappropriate to present the results on YouTube with a single real-valued label. For each model that is applicable to ISEAR, we show top 5 words of the generated label-specific topics in Table 4. It is worth to note that L-LDA and Dep-LDA achieve the same top words, since their difference only exists in the process of label prediction. The results indicate that although all models can learn meaningful topics, SLTM performs better than baseline models in label-specific topic discovery. For example, two words "happy" and "joy" which are strongly related to the label of "joy" are identified by SLTM with large probabilities. Similar results can be observed in other labels, thus topics discovered by our model are more convenient to be understood than others. Such a kind of performance enhancement is valuable to many real-world applications, e.g., personality education and psychotherapy, by producing human interpretable topics/events that evoke users' particular emotions.

For completeness, we also examine all topics generated by the baseline of sNTM. As mentioned earlier, sNTM is based on n-grams, instead of single words for SLTM and other baseline models. In the practical implementation, only unigrams and bigrams are considered since the embedding representation becomes less precise as n increases (Cao et al., 2015). The results indicate that sNTM can generate some topic bigrams such as "smelled disgusting" and "graduation exams", which are more appropriate to expressing a topic. However, only three topics are manually examined to be correlated with the seven emotions. This validates that sNTM is hard to introduce the guidance of labels in topic generation, because it models documents and labels separately.

To further evaluate the interpretability of topics extracted from SLTM, we firstly get topic embeddings by: $emb(z_k) = \mathbf{W}_1[k,:]$. Then, we map $emb(z_k)$ to a two-dimensional space via Principal Component Analysis (PCA). Figure 3 presents distributions of topics generated by SLTM over the ISEAR dataset. The scatter plot indicates that topics corresponding to the same label are closer than those of different labels. Furthermore, the distance between topics on correlated labels such as "fear" and "anger", is closer than that of topics on "joy" and other labels.

5.4 Evaluation on Label Prediction

We here evaluate the quality of word embeddings generated by SLTM on predicting categorical and real-valued labels based on ISEAR and YouTube, respectively. Since there are varied parameters for different models, we randomly select 60% of instances as the training set, 20% as the validation set, and the remaining 20% as the testing set. The values of parameters (e.g., the number of topics) for each model are all determined by the validation set. In label prediction, the main difference between SLTM and other supervised topic models is as follows. On one hand, a label-specific word embedding is introduced for predicting labels in SLTM according to Equation 10. On the other hand, other supervised topic models for both categorical and real-valued label prediction tasks infer labels for unlabeled documents by topic distributions directly, in which, topic distributions of unlabeled documents are learned without the supervision of labels.

For the task of categorical label prediction, the

Table 6: Regression performance on YouTube.

	MAE	pR^2
SVR	0.1424	-0.0591
HCNN	0.1112	0.3462
LDA+LR	0.1408	-0.0069
sLDA	0.1583	-0.2836
BP-sLDA	0.1394	-0.0208
sNTM	0.1342	0.0807
SLTM	**0.1005**	**0.4112**

accuracy and the Cohen's kappa score (Artstein and Poesio, 2008) are used as the evaluation metrics. Table 5 shows the classification performance of different models on ISEAR, where the best results are highlighted in boldface. For the prediction of real-valued labels on YouTube, we compare different models' regression performance by the mean absolute error (MAE) and the predictive R^2 (pR^2) (Blei and McAuliffe, 2007), as shown in Table 6. From the above results we can observe that SLTM achieves substantial performance improvement over baselines in predicting both categorical and real-valued labels, which indicates that word embeddings generated from labeled documents are more suitable for label prediction tasks than topic distributions generated from unlabeled documents without the guidance of labels.

5.5 Similarity of Word Embeddings

Word embeddings can reflect relations between words, and most methods of generating word embeddings are based on the local context information. This is because words with similar contexts may have similar semantics. However, a large-scale corpus is required to learn high quality word embeddings from the local context. Different from the previous word embedding generation methods, SLTM generates word embeddings based on the global label-specific topic information (i.e., the topical embedding space). Therefore, we further compare the quality of word embeddings learned by SLTM and three widely used methods: Word2Vec (W2V) (Mikolov et al., 2013), subword information Word2Vec (siW2V) (Joulin et al., 2017), and SSPMI (Levy and Goldberg, 2014). Among these baseline word embedding models, W2V and siW2V use the neural network framework, and SSPMI implicitly factorizes the pointwise mutual information (PMI) matrix of the local word co-occurrence patterns.

Table 7: Word similarity results on ISEAR.

	MEN	SimLex	Rare
W2V	0.002	-0.008	-0.119
siW2V	0.002	0.017	0.062
SSPMI	0.023	0.028	-0.004
SLTM	**0.169**	**0.037**	**0.089**

Table 8: Word similarity results on YouTube.

	MEN	SimLex	Rare
W2V	-0.018	0.004	-0.036
siW2V	-0.002	0.019	-0.051
SSPMI	-0.031	0.038	-0.026
SLTM	**0.048**	**0.040**	**0.068**

As our evaluation metric, the word similarity is estimated as follows. Firstly, we calculate cosine similarity scores for word pairs which occur in both the training set and the testing set. Secondly, word pairs are ranked according to their cosine similarities in the embedding space and human-assigned similarity scores, respectively. Finally, rankings of word similarity scores are evaluated by measuring the Spearman's rank correlation with rankings of human-assigned similarity scores. A higher correlation value indicates that it is more consistent to human judgements in word similarity. The following standard corpora which contain word pairs associated with human-assigned similarity scores are used for this evaluation: MEN (Bruni et al., 2014), SimLex-999 (SimLex) (Hill et al., 2015), and Rare (Luong et al., 2013).

We train W2V, siW2V, and SSPMI over each corpus by setting the number of context window size to 5. Furthermore, the dimension of word embeddings generated from all models is set to 50 according to (Lai et al., 2016). The values of word similarity on ISEAR and YouTube are respectively shown in Table 7 and Table 8, where the best results are highlighted in boldface. We can observe that SLTM outperforms baselines for all cases. The results indicate that word embeddings learned from the global label-specific topic information are better than those from the local context information without any external corpora.

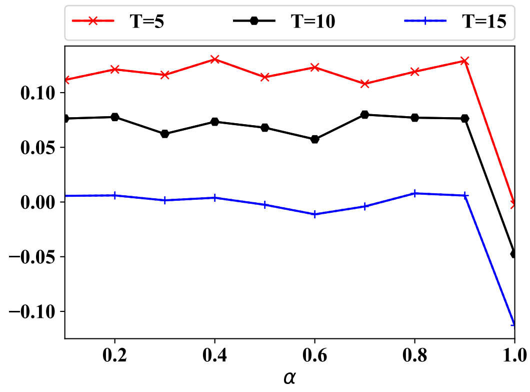

Figure 4: Topic coherence scores on ISEAR using different α values.

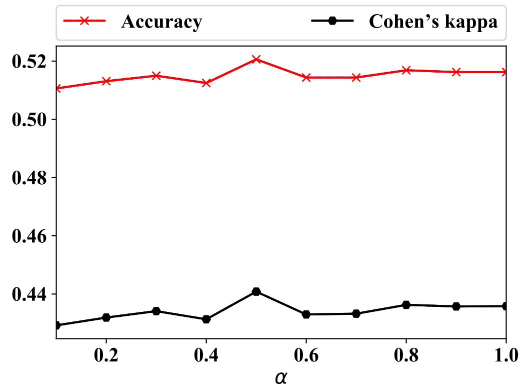

Figure 5: Label prediction performance on ISEAR using different α values.

5.6 Effect of the Hyper-parameter

After validating the effectiveness of SLTM on discovering topics and learning word embeddings, we now investigate the effect of the hyper-parameter in SLTM on these two aspects. According to Equation 8, the hyper-parameter α is used to weight two kinds of loss functions. Since $\mathbf{W}_2$ can be updated subject to $\alpha > 0$, we evaluate the performance of SLTM by varying α from 0.1 to 1 over the ISEAR dataset, as follows.

First, we evaluate the influence of hyper-parameter α on topic discovery by the coherence score of topics. To clearly illustrate the performance trend with different values of α, we set the number of top words T to 5, 10, and 15, and present topic coherence scores in Figure 4. The results indicate that SLTM performs stably under these α values on topic discovery, except for $\alpha = 1$ which ignores the label information totally. This validates the importance of label information in generating coherent topics.

Second, we use the learned word embeddings to predict document labels under different values of α. As shown in Figure 5, we can observe that when $\alpha = 0.5$, i.e., $loss(d_i, d_i^{(v_j-)})$ and $loss(\mathbf{y}_i, \mathbf{y}_i^{(v_j-)})$ are weighted equally, SLTM achieves the best performance in label prediction. The results indicate that the co-occurrence of documents and words as well as the label information are both important to generate good word embeddings. Furthermore, the label prediction performance of SLTM using any of these α values is better than that of most baselines (ref. Table 5). This validates the robustness of SLTM with different hyper-parameter values in supervised learning.

We also conduct experiments on YouTube using varied α values, which indicates that the hyper-parameter has a similar effect on both datasets.

6 Conclusion

In this paper, we proposed a supervised topic model named SLTM to discover label-specific topics by jointly modeling documents and labels. For the SLTM, weight matrices which represent document-topic and topic-word distributions can strictly follow probabilistic characteristics of topic models. Experiments were conducted on datasets with both categorical and real-valued labels, which validated that SLTM can not only discover more coherent topics, but also boost the performance of supervised learning tasks by learning high quality word embeddings. For future work, we plan to speed-up the training process of SLTM by GPUs and distributed algorithms. With the development of deep learning techniques, we also plan to de-emphasize irrelevant words with an attention mechanism.

Acknowledgments

We are grateful to the anonymous reviewers for their valuable comments on this manuscript. The research has been supported by the National Natural Science Foundation of China (61502545, U1711262, U1611264), a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (UGC/FDS11/E03/16), and the Innovation and Technology Fund (Project No. GHP/022/17GD) from the Innovation and Technology Commission of the Government of the Hong Kong Special Administrative Region.

References

Ron Artstein and Massimo Poesio. 2008. Inter-coder agreement for computational linguistics. *Computational Linguistics*, 34(4):555–596.

Yoshua Bengio. 2012. Practical recommendations for gradient-based training of deep architectures. In *Neural Networks: Tricks of the Trade - Second Edition*, pages 437–478. Springer-Verlag.

David M. Blei. 2012. Probabilistic topic models. *Communications of the ACM*, 55(4):77–84.

David M. Blei and Jon D. McAuliffe. 2007. Supervised topic models. In *Proceedings of the Annual Conference on Neural Information Processing Systems*, pages 121–128.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2001. Latent dirichlet allocation. In *Proceedings of the Annual Conference on Neural Information Processing Systems*, pages 601–608.

Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. 1993. Signature verification using a siamese time delay neural network. In *Proceedings of the Annual Conference on Neural Information Processing Systems*, pages 737–744.

Elia Bruni, Nam-Khanh Tran, and Marco Baroni. 2014. Multimodal distributional semantics. *Artificial Intelligence Research*, 49:1–47.

Ziqiang Cao, Sujian Li, Yang Liu, Wenjie Li, and Heng Ji. 2015. A novel neural topic model and its supervised extension. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, pages 2210–2216.

Huijun Chen, Xin Li, Yanghui Rao, Haoran Xie, Fu Lee Wang, and Tak-Lam Wong. 2017. Sentiment strength prediction using auxiliary features. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pages 5–14.

Jianshu Chen, Ji He, Yelong Shen, Lin Xiao, Xiaodong He, Jianfeng Gao, Xinying Song, and Li Deng. 2015. End-to-end learning of LDA by mirror-descent back propagation over a deep architecture. In *Proceedings of the Annual Conference on Neural Information Processing Systems*, pages 1765–1773.

Lei Cui, Dongdong Zhang, Shujie Liu, Qiming Chen, Mu Li, Ming Zhou, and Muyun Yang. 2014. Learning topic representation for SMT with neural networks. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 133–143.

Edouard Grave, Tomas Mikolov, Armand Joulin, and Piotr Bojanowski. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 427–431.

Ruidan He, Wee Sun Lee, Hwee Tou Ng, and Daniel Dahlmeier. 2017. An unsupervised neural attention model for aspect extraction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 388–397.

Yulan He, Chenghua Lin, and Harith Alani. 2011. Automatically extracting polarity-bearing topics for cross-domain sentiment classification. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 123–131.

Felix Hill, Roi Reichart, and Anna Korhonen. 2015. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41(4):665–695.

Thomas Hofmann. 1999. Probabilistic latent semantic analysis. In *Proceedings of the 15th Conference on Uncertainty in Artificial Intelligence*, pages 289–296.

Baotian Hu, Zhengdong Lu, Hang Li, and Qingcai Chen. 2014. Convolutional neural network architectures for matching natural language sentences. In *Proceedings of the Annual Conference on Neural Information Processing Systems*, pages 2042–2050.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 427–431.

Siwei Lai, Kang Liu, Shizhu He, and Jun Zhao. 2016. How to generate a good word embedding. *IEEE Intelligent Systems*, 31(6):5–14.

Jey Han Lau, David Newman, and Timothy Baldwin. 2014. Machine reading tea leaves: Automatically evaluating topic coherence and topic model quality. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 530–539.

Omer Levy and Yoav Goldberg. 2014. Neural word embedding as implicit matrix factorization. In *Proceedings of the Annual Conference on Neural Information Processing Systems*, pages 2177–2185.

Weifeng Li, Junming Yin, and Hsinchun Chen. 2018. Supervised topic modeling using hierarchical dirichlet process-based inverse regression: Experiments on e-commerce applications. *IEEE Transactions on Knowledge and Data Engineering*, 30(6):1192–1205.

Bing Liu. 2012. *Sentiment Analysis and Opinion Mining*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers.

Thang Luong, Richard Socher, and Christopher D. Manning. 2013. Better word representations with recursive neural networks for morphology. In *Proceedings of the 17th Conference on Computational Natural Language Learning*, pages 104–113.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Proceedings of the Annual Conference on Neural Information Processing Systems*, pages 3111–3119.

Daniel Ramage, David Leo Wright Hall, Ramesh Nallapati, and Christopher D. Manning. 2009. Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 248–256.

Timothy N. Rubin, America Chambers, Padhraic Smyth, and Mark Steyvers. 2012. Statistical topic models for multi-label document classification. *Machine Learning*, 88(1-2):157–208.

Klaus R. Scherer and Harald G. Wallbott. 1994. Evidence for universality and cultural variation of differential emotion response patterning. *Personality and Social Psychology*, 66:310–328.

Richard Socher, Andrej Karpathy, Quoc V. Le, Christopher D. Manning, and Andrew Y. Ng. 2014. Grounded compositional semantics for finding and describing images with sentences. *Transactions of the Association for Computational Linguistics*, 2:207–218.

Duyu Tang, Furu Wei, Nan Yang, Ming Zhou, Ting Liu, and Bing Qin. 2014. Learning sentiment-specific word embedding for twitter sentiment classification. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 1555–1565.

Tijmen Tieleman and Geoffrey Hinton. 2012. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural Networks for Machine Learning*, 4(2):26–31.

Chong Wang, David M. Blei, and Fei-Fei Li. 2009. Simultaneous image classification and annotation. In *Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1903–1910.

Zhongqing Wang and Yue Zhang. 2017. A neural model for joint event detection and summarization. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 4158–4164.

Bernard Weiner and Sandra Graham. 1990. Attribution in personality psychology. *Handbook of personality: Theory and research*, pages 465–485.

Cort J Willmott and Kenji Matsuura. 2005. Advantages of the mean absolute error (mae) over the root mean square error (rmse) in assessing average model performance. *Climate Research*, 30(1):79–82.

Chuan Yang, Lihe Zhang, Huchuan Lu, Xiang Ruan, and Ming-Hsuan Yang. 2013. Saliency detection via graph-based manifold ranking. In *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3166–3173.