Self-paced Compensatory Deep Boltzmann Machine for Semi-Structured Document Embedding

Shuangyin Li*
iPIN, Shenzhen, China.
shuangyinli@ipin.com

Rong Pan
School of Data and Computer Science,
Sun Yat-sen University, China.
panr@sysu.edu.cn

Jun Yan
Microsoft Research Asia.
junyan@microsoft.com

Abstract

In the last decade, there has been a huge amount of documents with different types of rich metadata information, which belongs to the Semi-Structured Documents (SSDs), appearing in many real applications. It is an interesting research work to model this type of text data following the way how humans understand text with informative metadata. In the paper, we introduce a Self-paced Compensatory Deep Boltzmann Machine (SCDBM) architecture that learns a deep neural network by using metadata information to learn deep structure layer-wisely for Semi-Structured Documents (SSDs) embedding in a self-paced way. Inspired by the way how humans understand text, the model defines a deep process of document vector extraction beyond the space of words by jointing the metadata where each layer selects different types of metadata. We present efficient learning and inference algorithms for the SCDBM model and empirically demonstrate that using the representation discovered by this model has better performance on semi-structured document classification and retrieval, and tag prediction comparing with state-of-the-art baselines.

1 Introduction

There have been massive documents in many web applications with the development of Internet. One kind of documents, which consists of the plain text and a group of metadata, are ubiquitous source of information. This type of documents can be called the semi-structured documents (SSDs) [Zhang *et al.*, 2009; Li *et al.*, 2013b; Soto *et al.*, 2015; Li *et al.*, 2016]. It is an important task to learn a good representation for a semi-structured document. The aim of document embedding is to generate such representations that are useful for document classification and retrieval tasks. When embedding the SSDs, besides the plain text in a document, the structured metadata such as authors and keywords in a paper and the director and actors in a film may contain rich information and can provide a great help to extract more meaningful document feature in a similar way to how human understand unstructured text. For example, when we are reading a textual description about a movie "A team of explorers travel through a wormhole in an attempt to ensure humanity's survival.", we would have no sense about what movie it refers. Then some keywords (one type of metadata) such as "Father daughter relationship, Space travel, Time paradox" can tell us that it is a Sci-Fi and father-daughter emotional movie. So these metadata would help us to extract more meaningful features about the text. Moreover, when another type of metadata "Christopher Nolan, Matthew McConaughey" is given, it is easy to know that this movie is "Interstellar (2014)". From this example, we can see that people could better understand unstructured text when provided hierarchical background knowledge. This is consistent with the behavior of the brain that people constantly add background knowledge to more accurately understand what he/she reads. This paced process shows an effective way to exact latent features for text, especially when different types of metadata are provided.

Thus, it would be better to utilize the metadata information when modeling the text. Recently, many models have been proposed to model the text as well as the metadata, such as [Li *et al.*, 2013b]. Li et al. propose a method to learn from the SSDs using directed topic model, where the major drawback is that it treats all the metadata equally, which ignores the fact that the metadata may belong to different types, and could have very different importances and concepts for the documents. It is obvious that the different types of metadata need to be considered gradually, which means that the types of metadata should be modeled into a hierarchical structure, just like the way of human learning that people would gradually take advantages of different auxiliary information in reading. Another challenge is that the order of different types of metadata need to be considered when we build this hierarchical structure. Similarly when reading a document, we may think of different types of metadata in different concept layers.

Thus, we propose a novel Self-paced Compensatory Deep Boltzmann Machine (SCDBM) that combines deep learning models with a latent topic model for semi-structured document embedding, in which each type of metadata is selected in different layers of a deep Boltzmann machine [Salakhutdinov and Hinton, 2009a] as compensatory information. The proposed SCDBM takes the corpus with the metadata as the input to obtain the documents' embeddings after an effective combination of bag-of-word information and the metadata

*Part of this work was performed at Microsoft Research Asia.

through a deep Boltzmann machine. We build the metadata information as one of teacher networks described in [Hu *et al.*, 2016], where the metadata can be used as one type of learning resources.

It has two principal contributions. First, through simulating the process of understanding text of human, this model can embed the documents with the benefits of integrating metadata information. Second, it provides an automatic method to handle different types of metadata in a self-paced way to add them into respective hidden layers in the deep Boltzmann machine, and has an efficient learning procedure to estimate the model parameters for each metadata-compensatory hidden layer.

In the experiment section, we will present the experimental results on two corpora, Wikipedia and IMDB, to show the performance of the proposed model on semi-structured document embedding for document classification and retrieval and tag prediction when comparing with state-of-the-art baselines.

2 Related Work

To date, there has been lots of work on developing document embedding using undirected graphical models or directed graphical models. As an undirected graphical model, the Replicated Softmax Model (RSM) [Salakhutdinov and Hinton, 2009b] can be used to model and extract low-dimensional latent semantic representations from unstructured collection of documents. Nitish et al. provide a type of deep Boltzmann machine, called the over-replicated Softmax model in [Nitish *et al.*, 2013] to extract distributed semantic representations from unstructured documents in a different way. Directed graphical models, such as Latent Dirichlet Allocation (LDA) [Blei *et al.*, 2003], CTM [Blei and Lafferty, 2005] and RATM [Li *et al.*, 2017], have been extensively used for building generative probabilistic models of the bag of words in a document. Several topic models have been proposed to take advantage of the metadata given in documents, such as Tag-Weighted Topic Model (TWTM) [Li *et al.*, 2013b], Tag-Weighted Dirichlet Allocation (TWDA) [Li *et al.*, 2013a], Author Topic Model (ATM) [Rosen-Zvi *et al.*, 2004], Labeled LDA [Ramage *et al.*, 2009], and so on. These models are useful for modeling sparse count data based on the assumption of bag-of-words, and provide methods to handle the word count representations in a document and to extract its latent topics.

Models based on deep neural networks have shown a powerful capability on image processing, speech recognition and many other areas [Hinton *et al.*, 2006]. Le Roux and Bengio has proved that adding hidden units yields strictly improved modeling power, and that restricted Boltzmann machines (RBMs) are universal approximators of discrete distributions to represent any distributions [Le Roux and Bengio, 2008]. Thus, each layer in the hierarchical (deep) architecture of deep Boltzmann machine (DBM) [Salakhutdinov and Hinton, 2009a] may provide a particular representation about the input data. However, there are few works investigated document modeling based on deep neural networks in semi-structured document embedding. In our paper, we take advantage of the capability of DBM to get the document representation by adding each selected type of the metadata information into the DBM architecture level by level.

The curriculum learning [Bengio *et al.*, 2009] provides a learning paradigm where a model is learned by gradually including more complex samples in the training process, which can be used to select the metadata to integrate the text understanding process layerwisely. Self-paced learning [Kumar *et al.*, 2010] [Jiang *et al.*, 2015] present a way to learn the priority ordering of each data point in the training process, which just matches the metadata selection process in our work. Thus, we take advantages of the self-pace learning to learn the ordering of different types of metadata selected for each layer.

3 Self-paced Compensatory Deep Boltzmann Machine

3.1 Notation

Consider a semi-structured corpus $\mathbf{D} = \{(\mathbf{w}_1, \mathbf{m}_1), ..., (\mathbf{w}_d, \mathbf{m}_d)\}$, where each 2-tuple $(\mathbf{w}, \mathbf{m})$ denotes a document, where $\mathbf{w}$ represents the bag-of-words representation of the document, and $\mathbf{m}$ represents the metadata in the document. $\mathbf{m} = (\mathbf{m}^{(1)}, \mathbf{m}^{(2)}, ..., \mathbf{m}^{(k)}, ..., \mathbf{m}^{(t)})$ where $\mathbf{m}^{(k)}$ denotes the k-th type of the metadata in the corpus and t is the number of the types of metadata in the corpus. Similar to deep Boltzmann machine, we define $\mathbf{h} = (\mathbf{h}^{(1)}, \mathbf{h}^{(2)}, ..., \mathbf{h}^{(i)}, \ldots, \mathbf{h}^{(t)})$ as the hidden layers in our model.

3.2 The proposed Model: SCDBM

To form a self-paced compensatory metadata structure, we gradually add the types of metadata layers as the compensatory information into the construction of deep Boltzmann machine layer by layer. Figure 1, left panel, shows an example of the proposed model with three hidden layers $\mathbf{h} = (\mathbf{h}^{(1)}, \mathbf{h}^{(2)}, \mathbf{h}^{(3)})$ and three types of metadata considered in a corpus where $\mathbf{m} = (\mathbf{m}^1, \mathbf{m}^2, \mathbf{m}^3)$. We build a deep multi-layers Boltzmann machine first, and then combine each hidden layer with metadata by learning a joint RBM. As shown in the left panel, Figure 1, we first select one type of metadata $\mathbf{m}^1$ as the compensatory metadata layer, then combine the input layer $\mathbf{w}$ to form an RBM with $\mathbf{h}^{(1)}$, where the compensatory metadata set $\langle \mathbf{m} \rangle = \{\mathbf{m}^1\}$. For the next level, we select another type of metadata to add to the compensatory metadata layer, where the set $\langle \mathbf{m} \rangle = \{\mathbf{m}^1, \mathbf{m}^2\}$, to combine with $\mathbf{h^{(1)}}, \mathbf{h^{(2)}}$ to form an RBM. For the last level, the remaining type of metadata is added to compensatory metadata layer $\mathbf{m}$, where the set $\langle \mathbf{m} \rangle = \{\mathbf{m}^1, \mathbf{m}^2, \mathbf{m}^3\}$, and then we learn an RBM to get the final hidden layer $\mathbf{h}^{(3)}$ as the embedding of $(\mathbf{w}, \mathbf{m})$.

Clearly, we need to build a t-layer of deep structure if there are t types of metadata in the corpus. Consider a t-layer SCDBM model. The energy of the state $\{\mathbf{w}, \mathbf{m}^1, ..., \mathbf{m}^t, \mathbf{h}^{(1)}, ..., \mathbf{h}^{(t)}\}$ is defined as (ignoring bias terms on the hidden units for clarity):

$$E(\mathbf{w}, \mathbf{m}^1, ..., \mathbf{m}^t, \mathbf{h}^{(1)}, ..., \mathbf{h}^{(t)}; \theta) = - \mathbf{w}^{\mathrm{T}} W^1 \mathbf{h}^{(1)} - \sum_{t}^{2} \mathbf{h}^{(t-1)\mathrm{T}} W^t \mathbf{h}^{(t)} - \sum_{t}^{1} |\mathbf{m}|^{t\mathrm{T}} \hat{W}^t \mathbf{h}^{(t)} \quad (1)$$

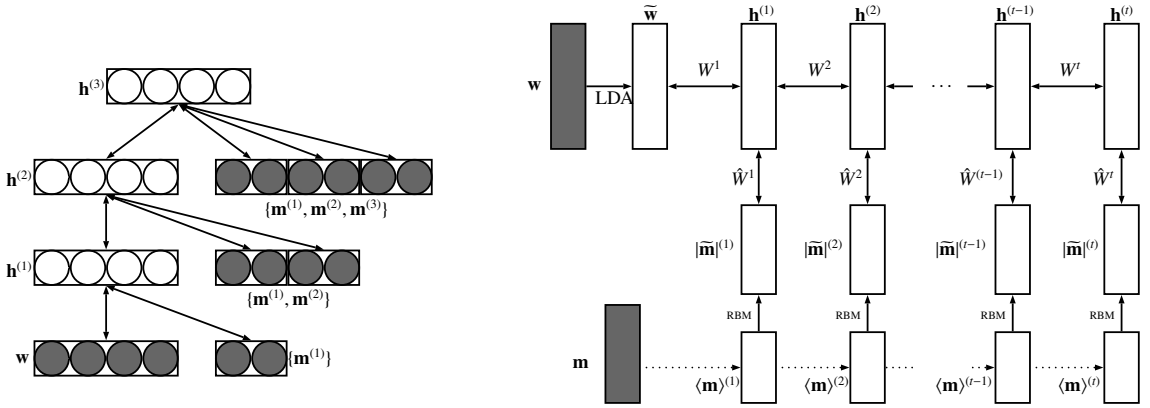


Figure 1: **Left** : A simple three-layer SCDBM with three types of metadata, in which the shadow nodes represent the observed variables: words and metadata in a document. **Right** : The graphical models of SCDBM by embedding words count vectors with latent topic model (LDA) and embedding metadata with RBMs.

where $|\mathbf{m}|^t = \mathbf{m}^1|\mathbf{m}^2|...|\mathbf{m}^t$ means we concatenate the types of metadata as the compensatory metadata layer. In the above equation, $\theta = \{W^1, ..., W^t, \hat{W}^1, ..., \hat{W}^t\}$ are the model parameters: W^1 and W^t are $\mathbf{w}$-to-$\mathbf{h}^{(1)}$ and $\mathbf{h}^{(t-1)}$-to-$\mathbf{h}^{(t)}$ symmetric interaction terms, and $\hat{W}^i$ is symmetric matrices of weights associated with the connections between hidden layers and compensatory metadata layers. The probability that the model assigns to word data $\mathbf{w}$ and metadata $(\mathbf{m}^1, ..., \mathbf{m}^t)$ is:

$$\begin{aligned} p(\mathbf{w}, \mathbf{m}^1, ..., \mathbf{m}^t; \theta) &= \sum_{\mathbf{h}^{(1)},...,\mathbf{h}^{(t)}} p(\mathbf{w}, \mathbf{m}^1, ..., \mathbf{m}^t, \mathbf{h}^{(1)}, ..., \mathbf{h}^{(t)}) \\ &= \frac{1}{Z(\theta)} \sum_{\mathbf{h}^{(1)},...,\mathbf{h}^{(t)}} \exp(-E(\mathbf{w}, \mathbf{m}^1, ..., \mathbf{m}^t, \mathbf{h}^{(1)}, ..., \mathbf{h}^{(t)}; \theta)), \end{aligned} \tag{2}$$

where $Z(\theta)$ is the partition function.

3.3 Self-paced Metadata Selection and Pretraining

As described above, the SCDBM selects the types of metadata layer-wisely to add into the deep structure. Then how to define the order of the types of metadata selected needs to be considered. The sequence of gradually added training samples, or other data, such as types of metadata can be called a curriculum mentioned in [Bengio *et al.*, 2009]. A straight forward way to select the sequence of metadata is to use metaheuristic measurements [Spitkovsky *et al.*, 2009]. While as described in [Kumar *et al.*, 2010], self-paced learning is a recently proposed learning regime inspired by the learning process of humans and animals that gradually incorporates data into training process. Actually, the proposed compensatory deep Boltzmann machine can be treated as one special self-paced learning problem where the main difference lies in the training data. Self-paced learning defines the order of training samples, and in our proposed model, the training samples are naturally separated into different data types, such as different types of metadata (the words in a semi-structured document can also be treated as one special type of metadata). Moreover, in the proposed model, we only consider the sequence of the different types of metadata where the number of types could be very small. The benefit is that we can easily learn the sequence through defining a problem-specific self-paced learning process. Thus, in the following, we introduce a tailored self-paced metadata selection process during pretraining layer-wisely.

We take advantage of a greedy layer-wise training process [Bengio *et al.*, 2007] based on learning a stack of "modified" RBMs using Contrastive Divergence (CD) algorithm [Hinton, 2002] for model pretraining and self-paced metadata selection. For the k-th layer, the input layer in the "modified" RBM consists of by two parts: the hidden layer $\mathbf{h}^{(k)}$ and the compensatory metadata layer $\mathbf{m}^{(k+1)}$, where the $\mathbf{m}^{(k+1)}$ contains $k+1$ selected types of metadata $\langle \mathbf{m} \rangle_{selected}$. The output layer is $\mathbf{h}^{(k+1)}$ and the candidate types of metadata set is denoted by $\langle \mathbf{m} \rangle_{left}$.

In order to select one type of metadata from $\langle \mathbf{m} \rangle_{left}$, we train the "modified" RBM with all the types of metadata from $\langle \mathbf{m} \rangle_{left}$ with $\mathbf{m}^{(k+1)}$ and $\mathbf{h}^{(k)}$, and the goal is to jointly learn the model parameters θ and a latent weight variable $\mathbf{v} = [v_1, ..., v_{t-k-1}]$ by maximizing the Pseudolikelihood of this "modified" compensatory RBM:

$$\begin{aligned} &\max_{\mathbf{v},\theta} \mathcal{L}(\mathbf{h}^{(k)}, \mathbf{h}^{(k+1)}, \langle \mathbf{m} \rangle_{selected}, \langle \mathbf{m} \rangle_{left}; \mathbf{v}, \theta) \\ &= \sum_{i}^{t-k-1} v_i \mathcal{L}(\mathbf{h}^{(k)}, \mathbf{h}^{(k+1)}, \langle \mathbf{m} \rangle_{selected}, \mathbf{m}^i_{left}) + \lambda \sum_{i}^{t-k-1} v_i, \end{aligned} \tag{3}$$

where λ is the parameter of learning pace and we set it to a constant here. We define $\mathbf{v}$ as an indicator vector that only one dimension equals 1, because we need to select one type of metadata. Unfortunately, jointly training the "modified" compensatory RBM with all the types of metadata involved using Eq. (3) would be very slow. Note that with the assumption of $\mathbf{v}$ as an indicator vector, the training process can be simplified, since we can search the space of $\mathbf{v}$ to find the solution, especially when the dimension of $\mathbf{v}$ is small in real. Thus, maximizing the Pseudolikelihood of this "modified" compensatory RBM is equivalent to maximizing the decision function:

$$\arg\max_{\mathbf{m}^i \in \langle \mathbf{m} \rangle_{left}} \mathcal{L}(\mathbf{h}^{(k)}, \mathbf{h}^{(k+1)}, \langle \mathbf{m} \rangle_{selected}, \mathbf{m}^i_{left}). \tag{4}$$

For the first layer, we use the $\mathbf{w}$ as the input layer and select one type of metadata as the compensatory layer by the above decision function. Note that, we only choose one type of metadata to add into the compensatory metadata set $\langle \mathbf{m} \rangle$

for each layer, so the depth of the proposed model is determined by the semi-structured corpus, where k types of metadata determine k layers of our model. The intuitive explanation is that the selection process is greedy, and the metadata which have more contribution to model the words in a document would be selected prior to that having less contribution. This process could mimic the human learning process that the human gradually take advantages of different auxiliary information when learning.

To train the "modified" compensatory RBM with each candidate type of metadata, the conditional distributions over $\mathbf{h}^{(k)}$, $\langle \mathbf{m} \rangle_{selected}$, $\mathbf{h}^{(k+1)}$ and the candidate $\mathbf{m}^c$ can be easily derived as follows:

$$p(\mathbf{h}_l^{(k+1)} = 1 | \mathbf{h}^{(k)}, \langle \mathbf{m}, \mathbf{m^c} \rangle) = g(\sum_p W_{pl}^k \mathbf{h}_p^{(k)} + \sum_q \hat{W}_{ql}^k |\mathbf{m}, \mathbf{m^c}|_q), \quad (5)$$

$$p(\mathbf{h}_p^{(k)} = 1 | \mathbf{h}^{(k+1)}) = g(\sum_l W_{pl}^k \mathbf{h}_l^{(k+1)}), \quad (6)$$

$$p(|\mathbf{m}, \mathbf{m^c}|_q = 1 | \mathbf{h}^{(k+1)}) = g(\sum_l \hat{W}_{ql}^k \mathbf{h}_l^{(k+1)}), \quad (7)$$

where $|\mathbf{m}, \mathbf{m^c}| = \mathbf{m}^1 | \mathbf{m}^2 | ... | \mathbf{m}^k | \mathbf{m}^c$, and $g(x) = 1/(1 + exp(-x))$ is the logistic function. We can also use rectified linear units described in [Nair and Hinton, 2010].

Metadata Embedding

In many applications, different types of metadata in one semi-structured corpus come from different kinds of information, such as the discrete and sparse category information, the real-valued and dense image pixel or timestamp information. This would make it difficult to learn the SCDBM model directly with the various forms of the metadata information without any preprocessing. Thus, we embed different kinds of metadata using RBMs (or their generalizations to exponential family models which have powerful representation as shown in [Le Roux and Bengio, 2008] and [Welling *et al.*, 2004]) as shown in Figure 1, right panel. We let $\widetilde{\mathbf{m}}$ denote the embedding of metadata. In particular, the traditional restricted Boltzmann machine could map the discrete and sparse data into low-dimensional representation, and the Gaussian RBM can be used to handle the real-valued data (e.g. image patches).

The main purpose that we do the metadata embedding is to reduce the dimensions of the metadata. When one type metadata is noisy, this would degrade the performance of the proposed model. Moreover, the observed metadata is typically high-dimensional in many applications, and if we use the observed metadata directly, the high-dimensional representation of $\mathbf{m}$ would make the model learning slowly. Another intuition behind the metadata embedding is that each type of metadata may have very different statistical properties that make it difficult to directly combine with hidden layers as compensatory information. The way of metadata embedding is to learn latent representations of metadata to remove such type-specific property so that the metadata information can be compensated to the corresponding hidden layers. On the other side, the metadata can be treated as one type of instance learning resources, or the simple rules that guide the building process of the network as shown in [Hu *et al.*, 2016]. The metadata embedding process may enhance the benefits of the metadata for performance improvement of text embedding.

Similar to the metadata, instead of directly using the word-count vectors of documents as input of the deep network, we use latent topic models (such as LDA or RSM) to extract the latent topic vectors before SCDBM learning. As shown in the right panel of Figure 1, $\widetilde{\mathbf{w}}$ represents the latent topic distribution obtained via a LDA or RSM, and then we use $\widetilde{\mathbf{w}}$ as the input of the first layer in SCDBM.

3.4 Approximate Learning and Inference

Exact maximum likelihood learning in this model is intractable. Thus, we carry out an efficient approximate learning process using the mean-field inference to estimate data-dependent expectations. [Salakhutdinov and Hinton, 2009a] describes a mean-field inference to estimate data-dependent expectations in DBM and an MCMC based stochastic approximation procedure to approximate the DBM's expected sufficient statistics. In this work, we use the same procedure for SCDBM model learning. Specifically the true posterior $p(\mathbf{h}|\widetilde{\mathbf{w}}, \widetilde{\mathbf{m}}; \theta)$ with a fully factorized approximating distribution over the t sets of hidden units is approximated as:

$$q(\mathbf{h}|\widetilde{\mathbf{w}}, \widetilde{\mathbf{m}}; \mu) = \left(\prod_i q(h_i^{(1)}|\widetilde{\mathbf{w}}, \widetilde{\mathbf{m}})\right)\left(\prod_i q(h_i^{(2)}|\widetilde{\mathbf{w}}, \widetilde{\mathbf{m}})\right) \cdots \left(\prod_i q(h_i^{(t)}|\widetilde{\mathbf{w}}, \widetilde{\mathbf{m}})\right) \quad (8)$$

where $\mu = \{\mu^{(1)}, ..., \mu^{(t)}\}$ are the mean-field parameters with $q(h_i^{(j)} = 1) = \mu_i^{(j)}$ for $j = 1, ..., t$. Thus, the variational lower bound on the log-probability of the data can be formulated as follows:

$$\begin{aligned} \log P(\widetilde{\mathbf{w}}, \widetilde{\mathbf{m}}; \theta) \geq\ & \mathbf{w}^{\mathrm{T}} W^1 \mu^{(1)} + |\widetilde{\mathbf{m}}|^{(1)^{\mathrm{T}}} \hat{W}^1 \mu^{(1)} \\ & + \sum_{i=1}^{t} \mu^{(i-1)^{\mathrm{T}}} W^i \mu^{(i)} + |\widetilde{\mathbf{m}}|^{(t)^{\mathrm{T}}} \hat{W}^t \mu^{(t)} - \log Z(\theta) + H(q), \end{aligned} \quad (9)$$

where $H(q)$ is entropy of the variational distribution and $|\widetilde{\mathbf{m}}|^{(t)}$ denotes the t-th layer's metadata concatenation. The learning process is as follows. First, we find the value of μ which maximizes the lower bound $\log P(\widetilde{\mathbf{w}}, \widetilde{\mathbf{m}}; \theta)$ given the current value of model parameters θ. Then, given the variational parameters μ, we can update the model parameters θ to maximize the variational bound using stochastic approximation [Salakhutdinov and Hinton, 2009a; Tieleman, 2008]. Actually, the number of types of metadata would be very small in real applications, and the training process is very efficient, especially after metadata embedding.

3.5 Metadata Prediction

After training the SCDBM model, we can infer the type of metadata compensated at the each level of the structure. For example, when we need to predict the metadata at layer i which means the $i+1$ type of metadata in the compensatory sequence is missing, $\mathbf{m}^{i+1}$, we first infer the values of the hidden variables $\mathbf{h^{(i)}}$ with $\mathbf{w}$ and the metadata sets $\langle \widetilde{\mathbf{m}} \rangle^{(i)}$, then we can perform alternating Gibbs sampling using the following conditional distributions until convergence:

$$p(\mathbf{h^{(i+1)}}|\mathbf{h^{(i)}}, \langle \widetilde{\mathbf{m}} \rangle^{(i)}, \widetilde{\mathbf{m}}^{i+1}) = g(W^i \mathbf{h^{(i)}} + \hat{W}_a^i \widetilde{\mathbf{m}}^{(i)} + \hat{W}_b^i \widetilde{\mathbf{m}}^{i+1}), \quad (10)$$

$$p(\widetilde{\mathbf{m}}^{i+1}|\mathbf{h^{(i+1)}}) = g(\hat{W}_b^i \mathbf{h^{(i+1)}}), \quad (11)$$

where W^i is the weight matrix between $\mathbf{h^{(i)}}$ and $\mathbf{h^{(i+1)}}$, and $\hat{W}_b^i$ is the weight matrix between $\mathbf{h^{(i+1)}}$ and $\widetilde{\mathbf{m}}^{(i+1)}$. We use the distribution of $p(\widetilde{\mathbf{m}}^{i+1}|\mathbf{h^{(i+1)}})$ to sample $\widetilde{\mathbf{m}}^{i+1}$. The sample $\widetilde{\mathbf{m}}^{\mathbf{(i+1)}}$ can then be propagated back to generate a distribution over the vector space of metadata $\mathbf{m^{(i+1)}}$, which can be used to sample the real metadata.

For the self-paced metadata selection process when pretraining, the complexity is same as a traditional RBM in each layer for one type of metadata selection. Thus, the total complexity in self-paced metadata selection process is t times for a traditional RBM, where t is the number of type of metadata (both the depth of the network). For approximate learning process, the complexity is same as a traditional DBM, since the compensatory metadata for each layer has fixed.

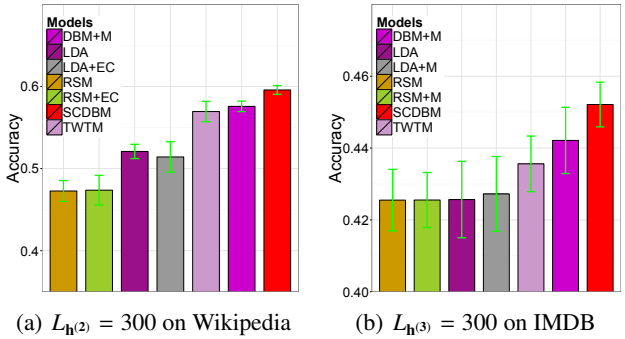
4 Experiments

4.1 Description of Datasets

In the experiments, we used two semi-structured corpora. The first dataset is from Wikipedia. The Wikipedia corpus used in our experiment contains D = 213,600 articles, with W = 70,061 words in the vocabulary by removing the stop words. Each article contains two types of metadata. The first type of the metadata $\mathbf{m}^1$ is the entity information observed in the abstract of the article with the hyper links. The $\mathbf{m}^2$ is the category information normally found at the bottom of an article page. There are 974 categories in the category dictionary and 10,208 entities after removed low-frequency items. Each article belongs to one of 18 classes, such as history, education, technology, art and so on. The second corpus is the data from Internet Movie Database (IMDB). The data set we used includes 106,432 movie story lines, and 15,703 words after removing the stop words and low frequency words. Each movie contains three types of metadata. a) $\mathbf{m}^1$, the crew of the movie including the director, the writers and the actors. b) $\mathbf{m}^2$, a list of keywords. c) $\mathbf{m}^3$, the movie information including rating, production, country, language, the length of duration, sound mix and date. For $\mathbf{m}^3$, we simply used a 6 dimension real-value vector to represent. There are 5,192 persons appeared in all the movie by removing the persons appeared less than 5 times, and 5,773 keywords in the keyword dictionary. Each movie belongs to one of 29 genres.

4.2 Experimental Setting

We compared the SCDBM with LDA, Replicated Softmax Model (RSM), and TWTM [Li *et al.*, 2013b]. Because LDA and RSM are the models handling the unstructured documents without the metadata information, so we treated the given metadata as word features for them. For the Wikipedia, we treated all the metadata, including entities and categories, as the words to train LDA and RSM, which are called LDA-EC and RSM-EC, respectively. Likewise, we also trained LDA and RSM on the IMDB with the metadata as the words features, and we called the models LDA-M and RSM-M. For the RSM model, we subdivided datasets into minibatches, and each contains 100 training cases, and updated the parameters after each minibatches to speed-up learning in the same way as RSM. Moreover, we choose the 2,500 most frequent words as the feature representations in the training dataset. With the three models, we embedded the documents into 300-dimensional latent representations.



(a) $L_{\mathbf{h^{(2)}}} = 300$ on Wikipedia (b) $L_{\mathbf{h^{(3)}}} = 300$ on IMDB

Figure 2: Classification results on the Wikipedia (a) and IMDB (b) for RSM, LDA, TWTM and SCDBM with 5-fold cross-validation.

When training the proposed SCDBM, we first embedded the metadata of the two corpora into low dimension using RBMs. For Wikipedia, we let $L_{\widetilde{\mathbf{m}}^{(1)}} = 300$, and $L_{\widetilde{\mathbf{m}}^{(2)}} = 300$. For IMDB, we let $L_{\widetilde{\mathbf{m}}^{(1)}} = 300$, $L_{\widetilde{\mathbf{m}}^{(2)}} = 300$ and $L_{\widetilde{\mathbf{m}}^{(3)}} = 6$. To model the word count vector, we trained LDAs with 300 topics on the two corpora, and treated the latent topic distributions as $\widetilde{\mathbf{w}}$ at the bottom level of our model. The dropout rate is set to 0.5 as described in [Srivastava *et al.*, 2014] when training our model.

Since the number of types of metadata in each corpus is small, the starting value of $\mathbf{v}$ is becomes critical. Thus, after tried some methods to initialize the $\mathbf{v}$, we use a logarithmic scheme to initialize the $\mathbf{v}$ as shown as in [Jiang *et al.*, 2015] to get the best results. After pretraining and selection for the types of metadata to be added into the compensatory metadata set, we obtained the following sequence for each corpora. For Wikipedia, we built a two-layer network with $\langle \mathbf{m}^1 \rangle^{(1)}$, $\langle \mathbf{m}^1, \mathbf{m}^2 \rangle^{(2)}$ as the compensatory metadata layers, and let $L_{\mathbf{h^{(1)}}} = L_{\mathbf{h^{(2)}}} = 300$. For IMDB, we built a three-layer network with $\langle \mathbf{m}^2 \rangle^{(1)}$, $\langle \mathbf{m}^2, \mathbf{m}^1 \rangle^{(2)}$, $\langle \mathbf{m}^2, \mathbf{m}^1, \mathbf{m}^3 \rangle^{(3)}$ as the compensatory metadata layers, and let $L_{\mathbf{h^{(1)}}} = L_{\mathbf{h^{(2)}}} = L_{\mathbf{h^{(3)}}} = 300$.

4.3 Document Classification and Retrieval

In our first set of experiments, we evaluated the single-class classification performance utilizing feature sets generated by SCDBM and other baselines on the Wikipedia and IMDB. As we have mentioned, we used the latent features of the semi-structured documents dataset embedded by SCDBM to train a classifier based on LIBSVM with the Gaussian kernel and the default parameters. Figure 2 shows the classification results on the two corpora with mean square errors. Intuitively, SCDBM takes full advantage of metadata, especially with the depth increases. In order to test the way of adding the different types of metadata into different hidden layers respectively, we also trained a variant model with adding all the metadata into the lowest level of a DBM, which we called DBM+M, compared with the self-paced metadata-compensatory way proposed in this paper. These comparisons indicate that the way of self-paced metadata-compensatory could bring the performance improvement. The main reasons are following. First, different types of metadata would have different concept properties, the way of self-paced selection in our model is just to find a better way to take advantages of this metadata. Second, under the setting of SCDBM, the metadata which is

Table 1: Some examples of the predictive categories generated by SCDBM on Wikipedia, where Column 2 is the real categories appeared in the test pages, and Column 3 in red is the highly correlated categories generated by SCDBM.

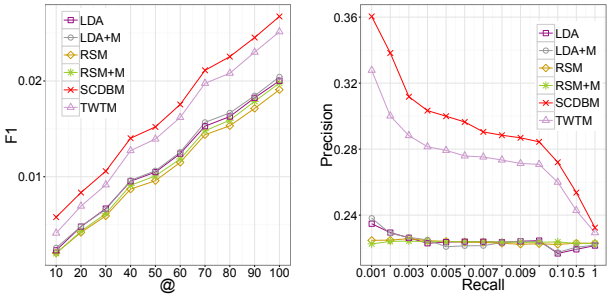
Article Titles	The categories predicted by SCDBM	
"Bubble Bobble"	Mobile games, Amiga games, Apple II games, Platform games	*Video games*, *Video games graphics*, *Game boy advance*
"Andrew Wiles"	MacArthur Fellows, Living people, 21st-century mathematicians	*People from London*, *Science*, *Fellows of the American Academy of Arts and Sciences*
"Leigh Brackett"	20th-century women writers, American screenwriters	*Films based on novels*, *Fantasy films*, *American science fiction horror films*, *Science fiction films*
"Furman University"	Educational institutions established in 1826	*Fellows of the American Academy of Arts and Sciences*, *Science*, *Education*
"NCAA football"	College football	*Middle States Association of Colleges and Schools*, *Sports*, *American football quarterbacks*, *Education in the United States*, *American football running backs*
"David Beckham"	English footballers, UEFA Euro 2004 players	*FIFA World Cup players*, *Scottish footballers*, *American films*, *National Football League announcers*, *Footballers in Italy*

selected in the first layer could influence the structure to the last layer, which means that the different types of metadata have different weights during the training.

We also used IMDB to evaluate the performance on a document retrieval task. To decide whether a retrieved document is relevant to the query document, we just check whether they have the same class label. In order to do retrieval, we represent each document $\mathbf{w}$ using the latent vector representation generated by SCDBM and other models in comparison. The corpus was randomly divided into two part: 80% the database documents and 20% the query documents. For each query, documents in the database were ranked by using the cosine distance as the similarity metric. We averaged the results of all the queries by using the F-Measure (F1-score). As shown in Figure 3(a), the SCDBM outperforms other models on F1-score. We also show the performances of different models on document retrieval by drawing the precision-recall curves. Figure 3(b) shows the results, from which the SCDBM also has a better performance than other baselines, particularly when retrieving the top few documents.

4.4 Results on Metadata Prediction

In this experiment, we show the generative aspect of our model qualitatively. We randomly selected 1,360 documents from the Wikipedia dataset as the test set, and treated the left 10,000 documents as the training set. Before training, we embedded the entities and categories of each document into low-dimensional space using the RBMs. When inferring a document, we only used the words ($\mathbf{w}$) and entities ($\mathbf{m}^{(1)}$)



(a) F1-Score (b) Precision-Recall curves

Figure 3: (a) F1-Score for document retrieval on IMDB. (b) Precision-Recall curves for document retrieval on IMDB.

of the document to generate the vector $\mathbf{h}^{(1)}$, then generated the distribution $\widetilde{\mathbf{m}}^{(2)}$ by the method which we mentioned in Section 3. Since the value of the unit in $\widetilde{\mathbf{m}}^{(2)}$ is the distribution of $p(\mathbf{m}^{(2)} = 1)$, We simply ranked the values in $\widetilde{\mathbf{m}}^{(2)}$ and selected the top 20 as the predictive categories generated by SCDBM. Table 1 shows some examples of the predictive categories generated by the SCDBM for a group of pages in the test set. The first column shows the articles in test set we inferred, and columns 2 and 3 are the categories predicted by our model; Column 2 is the real categories appeared in the test pages, and Column 3 (red) is the highly correlated categories generated by SCDBM. By taking the entity of "David Beckham" for an example, the categories of "English footballers" and "UEFA Euro 2004 players" are generated by our model, which are the real categories given in the article of "David Beckham". The categories (red) in Table 1 are highly correlated with the input article, even though some of them are wrong in fact. In other words, the SCDBM may have the capability of learning the internal relations among the metadata items, as well as capturing relation between words and metadata.

5 Conclusion

We proposed a self-paced compensatory deep Boltzmann machine to model documents by combining the word features and metadata information by simulating the process of people understanding text. The model fused the word count vector and metadata into a jointly hidden representation, and provided improved capability in terms of classification and document retrieval compared to state-of-the-art models. Besides, the model has demonstrated the ability to predict the missing metadata in some applications.

Acknowledgments

This work was supported in part by the National Key Research and Development Program of China under Grant 2016YFB0201900. Also, this work was supported in part by NSF of China under Grant 61672548, U1611461, and the Guangzhou Science and Technology Program, China, under Grant 201510010165. We thank the support of Key Laboratory of Machine Intelligence and Advanced Computing (Sun Yat-sen University), Ministry of Education, China.

References

[Bengio *et al.*, 2007] Yoshua Bengio, Pascal Lamblin, Dan Popovici, and Hugo Larochelle. Greedy layer-wise training of deep networks. In *NIPS*, 2007.

[Bengio *et al.*, 2009] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*. ACM, 2009.

[Blei and Lafferty, 2005] David M. Blei and John D. Lafferty. Correlated topic models. In *NIPS*, 2005.

[Blei *et al.*, 2003] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 2003.

[Hinton *et al.*, 2006] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 2006.

[Hinton, 2002] Geoffrey E Hinton. Training products of experts by minimizing contrastive divergence. *Neural computation*, 2002.

[Hu *et al.*, 2016] Zhiting Hu, Xuezhe Ma, Zhengzhong Liu, Eduard Hovy, and Eric Xing. Harnessing deep neural networks with logic rules. *arXiv:1603.06318*, 2016.

[Jiang *et al.*, 2015] Lu Jiang, Deyu Meng, Qian Zhao, Shiguang Shan, and Alexander G Hauptmann. Self-paced curriculum learning. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.

[Kumar *et al.*, 2010] M Pawan Kumar, Benjamin Packer, and Daphne Koller. Self-paced learning for latent variable models. In *NIPS*, 2010.

[Le Roux and Bengio, 2008] Nicolas Le Roux and Yoshua Bengio. Representational power of restricted boltzmann machines and deep belief networks. *Neural Computation*, 2008.

[Li *et al.*, 2013a] Shuangyin Li, Guan Huang, Ruiyang Tan, and Rong Pan. Tag-weighted dirichlet allocation. In *2013 IEEE 13th International Conference on Data Mining (ICDM)*, pages 438–447, 2013.

[Li *et al.*, 2013b] Shuangyin Li, Jiefei Li, and Rong Pan. Tag-weighted topic model for mining semi-structured documents. In *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence (IJCAI)*, 2013.

[Li *et al.*, 2016] Shuangyin Li, Rong Pan, Yu Zhang, and Qiang Yang. Correlated tag learning in topic model. In *Proceedings of the Thirty-Second Conference on Uncertainty in Artificial Intelligence*, pages 457–466, 2016.

[Li *et al.*, 2017] Shuangyin Li, Yu Zhang, Rong Pan, Mingzhi Mao, and Yang Yang. Recurrent attentional topic model. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17)*, 2017.

[Nair and Hinton, 2010] Vinod Nair and Geoffrey E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning*, 2010.

[Nitish *et al.*, 2013] Srivastava Nitish, Ruslan Salakhutdinov, and Geoffrey E Hinton. Modeling documents with a deep boltzmann machine. In *Uncertainty in Artificial Intelligence*, 2013.

[Ramage *et al.*, 2009] Daniel Ramage, David Hall, Ramesh Nallapati, and Christopher D. Manning. Labeled lda: A supervised topic model for credit attribution in multi-labeled corpora. In *EMNLP*, 2009.

[Rosen-Zvi *et al.*, 2004] Michal Rosen-Zvi, Thomas L. Griffiths, Mark Steyvers, and Padhraic Smyth. The author-topic model for authors and documents. In *UAI*, 2004.

[Salakhutdinov and Hinton, 2009a] Ruslan Salakhutdinov and Geoffrey E Hinton. Deep boltzmann machines. In *International Conference on Artificial Intelligence and Statistics*, 2009.

[Salakhutdinov and Hinton, 2009b] Ruslan Salakhutdinov and Geoffrey E Hinton. Replicated softmax: an undirected topic model. In *NIPS*, 2009.

[Soto *et al.*, 2015] Axel J Soto, Ryan Kiros, Vlado Kešelj, and Evangelos Milios. Exploratory visual analysis and interactive pattern extraction from semi-structured data. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 2015.

[Spitkovsky *et al.*, 2009] Valentin I. Spitkovsky, Hiyan Alshawi, and Daniel Jurafsky. Baby steps: How "less is more" in unsupervised dependency parsing. *Nips Grammar Induction Representation of Language and Language Learning*, 2009.

[Srivastava *et al.*, 2014] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 2014.

[Tieleman, 2008] Tijmen Tieleman. Training restricted boltzmann machines using approximations to the likelihood gradient. In *Proceedings of the 25th international conference on Machine learning*. ACM, 2008.

[Welling *et al.*, 2004] Max Welling, Michal Rosen-Zvi, and Geoffrey E Hinton. Exponential family harmoniums with an application to information retrieval. In *NIPS*, 2004.

[Zhang *et al.*, 2009] Duo Zhang, Chengxiang Zhai, and Jiawei Han. Topic cube: Topic modeling for olap on multidimensional text databases. In *Proceedings of the SIAM International Conference on Data Mining*, 2009.