

# Aspect-based Sentiment Analysis using Word Embedding Restricted Boltzmann Machines

Bao-Dai Nguyen-Hoang, Quang-Vinh Ha, and Minh-Quoc Nghiem

Faculty of Information Technology  
Ho Chi Minh City University of Science  
227 Nguyen Van Cu St., Ward 4, District 5, Ho Chi Minh City, Vietnam  
{1212069, 1212304}@student.hcmus.edu.vn, nqminh@fit.hcmus.edu.vn

**Abstract.** Recent years, many studies have addressed problems in sentiment analysis at different levels, and building aspect-based methods has become a central issue for deep opinion mining. However, previous studies need to use two separated modules in order to extract aspect-sentiment word pairs, then predict the sentiment polarity. In this paper, we use Restricted Boltzmann Machines in combination with Word Embedding model to build the joined model which not only extracts aspect terms appeared and classifies them into respective categories, but also completes the sentiment polarity prediction task. The experimental results show that the method we use in aspect-based sentiment analysis tasks is better than other state-of-the-art approaches.

**Keywords:** Aspect-based, Sentiment Analysis, Opinion Mining, Restricted Boltzmann Machine, Supervised Learning, Word Embedding

## 1 Introduction

Sentiment Analysis (also known as opinion mining) is the process of determining whether a piece of writing is positive or negative. With the development of opinionated user-generated review sites, many customers can write reviews and express their opinions about the products (or services). Sentiment Analysis could help not only users to choose the right products but also companies to improve their products based on these reviews.

Aspect-based Sentiment Analysis (ABSA) has received much attention in recent years since each review might contain many aspects. For example, in a restaurant review, we may have opinions about *food*, *staff*, *ambience*, etc. Conventional ABSA systems normally have two separated modules: one for aspect extraction and another one for sentiment classification [1–3]. Recently, Wang et al. [4] introduced a joint model, called Sentiment-Aspect Extraction based on Restricted Boltzmann Machines (SERBM), that extract aspects and classify sentiments at the same time. In this model, they used unsupervised Restricted Boltzmann Machine (RBM) and three different types of hidden units

to represent aspects, sentiments, and background information, respectively. Furthermore, they added prior knowledge into this model to help it acquire more accurate feature representations. The visible layer  $\mathbf{v}$  of SERBM is represented as a  $K \times D$  matrix, where  $K$  is the dictionary size and  $D$  is the document length. They showed that their model is well-suited for solving aspect-based sentiment analysis tasks.

However, there are two main problems still exist in the SERBM model. Firstly, an unsupervised method can only cluster reviews into categories and we can not know the name of the category. No information was given to determine which position of the hidden units to represent aspects, sentiments or background words during the training process. Secondly, a visible layer will be a matrix combined by high-dimensional vectors if training data has a large set of vocabulary, which requires much computational resource.

In this paper, we propose combining Restricted Boltzmann Machine with Word Embedding model to overcome the limitations of existing method. Word Embedding model has the capability of reducing the dimensionality of the input vectors. Therefore, we can use it to reduce the dimensionality of the input in visible layer while keeping the semantics of the reviews. We encode the input document as a vector, created by the Word Embedding model, instead the vector of one-hot encoding Bag Of Words model. Furthermore, we use RBM in a supervised setting. We move the output component from hidden layer to visible layer. The hidden layer now acts as the dependencies between the components in the visible layer. Doing like this, we can fix the units for desired categories.

We call our model Word Embedding Restricted Boltzmann Machine (WE-RBM). Overall, our main contributions are as follows:

- This is the first work that combines Word Embedding model and supervised RBM for the ABSA task. Compared with other state-of-the-art methods, our model can identify aspects and sentiments efficiently, yielding 1-6% improvements in accuracy for sentiment classification task and 1.73% to 7.06% improvements in F1 score for aspect extraction task.
- By using Word Embedding model, we can efficiently reduce the size of input vectors up to 100 times, which in turn reduces the training time greatly.
- We also introduce a simple yet efficient way to incorporate prior knowledge into RBM model. Prior knowledge is the advantage of Word Embedding model, which can help RBM to be well-suited for solving aspect-based opinion mining tasks.

The rest of this paper is organized as follows. Section 2 introduces the related work. Section 3 overviews the background information, then describes our approach to classify reviews into aspect categories and predict sentiment polarity of the reviews. Experimental results are presented in Section 4. Finally, Section 5 concludes the paper and discusses future work.

## 2 Related work

ABSA approaches may be divided into three main categories: rule-based, supervised learning, and unsupervised learning.

Rule-based approaches [1, 5] can perform quite well in a large number of domains. They use a sentiment lexicon, expressions, rules of opinions, and the sentence parse tree to help classify the sentiment orientation on each aspect appeared in a review. They also consider sentiment shifter words (i.e. *not*, *none*, *nobody*, etc.). However, these rule-based methods have a shortcoming in processing complex documents where the aspect is hidden in the sentence, and failing to group extracted aspect terms into categories.

For the supervised learning approach, Wei and Gulla [6] propose a hierarchical classification model to determine the dependency and the other relevant information in the sentence. Jiang et al. [7], Boiy and Moens [8] use dependency parser to generate a set of aspect-dependent features for classification, which weighs each feature based on the position of the feature relative to the target aspect in the parse tree. Several other supervised learning models have been published, such as Hidden Markov Models [9], SVMs [10], Conditional Random Fields [11, 12].

It has been demonstrated that a classifier trained from labeled data in one domain often performs poorly in another domain [13]. Hence, unsupervised methods are often adopted to avoid this issue. Several recent studies investigate statistical topic models which are unsupervised learning methods. They assume that each document consists of a mixture of topics. Specifically, Latent Dirichlet Allocation (LDA) [14], Multi-Grain LDA model [15] are used to model and extract topics from document collections. A number of authors have considered the effects of topic models on ABSA task, such as the two-step approach [16], joint sentiment/topic model [17], and topic-sentiment mixture model [18]. A recent study by Wang et al. [4] proposed the SERBM model which also jointly address these two tasks in an unsupervised setting.

## 3 Proposed method

### 3.1 Background

**Restricted Boltzmann Machine** RBM model, a generative stochastic artificial neural network, is treated as a model in the field of deep learning. RBM can be used to learn important aspects of an unknown probability distribution based on samples from this distribution [19]. Recently, researchers have applied RBM in the field of Natural Language Processing, including topic modeling and sentiment analysis [4]. One special characteristic of RBM is that it can be used in both supervised and unsupervised ways, depending on the task.

As shown in Figure 1, RBM model is a two-layer neural network which contains one visible layer and one hidden layer. The visible layer is constituted by visible units correspond to the components of an observation (e.g., one visible unit for each word in an input document). The hidden layer composed of hidden

units which model the dependencies between the components of observations (e.g., dependencies between words in the document).

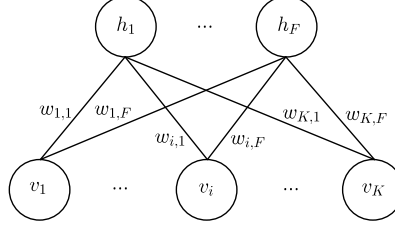


Fig. 1: The network graph of an RBM model with  $K$  visible and  $F$  hidden units

**Word Embedding model** The Word Embedding model (WEM) is a proven and powerful paradigm in Natural Language Processing, in which words are represented as vectors in a high-dimensional space [20]. The idea of this model is representing words as vectors. Therefore, the similarity of two words  $i$  and  $j$  can be calculated based on the cosine of the angle between the vectors as shown in equation 1.

$$\cos \theta = \frac{w_i \cdot w_j}{||w_i|| ||w_j||} \quad (1)$$

where  $w_i \cdot w_j$  is the dot product of these two documents while  $||w_i||$  and  $||w_j||$  are the norm of vector  $w_i$  and  $w_j$ , respectively.

For document representation, every word appeared in the document is represented as a vector. We then sum all of these vectors to get a vector that represent the document [20].

### 3.2 Our Aspect-based Sentiment Analysis model

**Structure** In the previous approach, Wang et al. [4] proposed an unsupervised RBM model to ABSA. As mention before, there are two main shortcomings in this SERBM model. Firstly, an unsupervised method can only cluster reviews into categories and we can not know the name of the category. This model fixes hidden units 0–6 to represent the target aspects *Food*, *Staff*, *Ambience*, *Price*, *Ambience*, *Miscellaneous*, and *Other Aspects*, respectively. There is no way we can determine the aspects (e.g. which unit represents *Food*, which unit represents *Staff*) based on the position of the hidden units alone.

Secondly, when training, each document is transformed into a  $K \times D$  matrix  $\mathbf{v}$ , where  $K$  is the dictionary size, and  $D$  is the document length. If visible unit  $i$  in  $\mathbf{v}$  takes the  $k$ -th value,  $v_i^k$  is set to 1. If the training data has a large set of vocabulary, the visible layer will be a matrix combined by high-dimensional

vectors. These sparse input vectors lead to not only the decrease in the model’s accuracy but also the increase in computational resources.

To overcome these problems, we propose a method using supervised RBM model which is illustrated in Figure 2. The **Output units** include the units represent the aspects and sentiment orientations of the reviews. We call them *Aspects identifying units* and *Sentiments identifying units*, respectively. These units are put together with **Input units** in the visible layers, instead of being placed in hidden layers. Meanwhile, the units in the hidden layer represent the relationship between the units in the visible layers. We encode input units in the visible layer as a vector, created by the Word Embedding model [21], instead the vector of one-hot encoding Bag Of Words model. This helps reduce the dimensionality of the input matrix while keeping documents’ semantics.

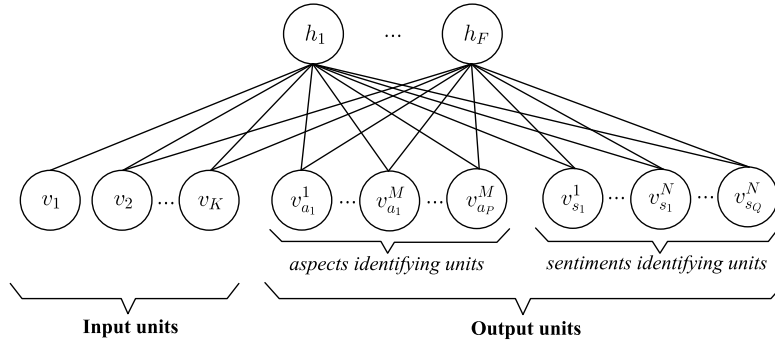


Fig. 2: The network graph of our Supervised Sentiment-Aspect Extraction RBM model

Compared to standard RBMs, the first difference of this model is the visible layer. Apart from the input units, there are also aspect and sentiment identifying units that represent the output component of the model. Suppose in the training data, there are reviews talking about  $P$  aspects and having  $Q$  sentiment orientations in total. In particular, the set of aspects is  $A = \{a_1, a_2, \dots, a_P\}$ , and the set of sentiment orientations is  $S = \{s_1, s_2, \dots, s_Q\}$ .

For each aspect, the model set  $M$  units to capture that aspect. For example, if the review mentioned aspect  $a_i \in A$ , we set the units from  $v_{a_i}^1$  to  $v_{a_i}^M$  to 1, and 0 otherwise. We use the same setting for sentiment units. The model set  $N$  units to capture each sentiment polarity. If the sentiment polarity is  $s_j \in S$ , we set the units from  $v_{s_j}^1$  to  $v_{s_j}^N$  to 1, and 0 otherwise. The idea of setting  $M$  and  $N$  units to capture aspects and sentiment polarity is obtained from previous work [19]. Our model has the property that input units and output units stay in the same layer. If we used only one output unit for each aspect and sentiment polarity while there are too many input units, the model would be imbalance and need more time to converge.

With this structure, our model can solve two tasks simultaneously: aspect identification and sentiment classification. This ability of our model is reflected

in the aspect and sentiment identifying units which play important roles in the sampling process of RBM. Meanwhile, weight values of connecting edges contain semantic information of the review, which help identifying units to communicate with the hidden layer. In addition, the fixed dimension of the input vector does not depend on the number of vocabulary words. This capability not only helps the model prevent the occurrence of decreasing speed and accuracy, but also solves the semantic problem in the review.

For the customer restaurant reviews analyzing task, one word in a review may mention a certain aspect (e.g. “delicious” corresponds to *food* aspect), or a certain opinion (e.g. “good” is about positive sentiment, while “bad” is about negative sentiment). Furthermore, there are other words do not mention about aspect or sentiment, they can be removed during the preprocessing phase. These latent topics in a review are considered as the factors which generate aspect and sentiment words within that review. To technically illustrate this, hidden layer contains information of the latent topics. In the generation process, hidden units produce the values of visible units, which are also the information of the aspect and sentiment words in reviews. Moreover, we can increase the number of hidden units in order to increase the modeling capacity of the RBM, which make the model powerful enough to represent complicated distributions.

**Word Embedding Restricted Boltzmann Machine** When we use supervised RBM with input vectors generated by Word Embedding model, the results showed that RBM does not have enough capacity to regenerate visible units. Much of the instability in this approach stems from two reasons. Firstly, RBM has only hidden and visible layers to capture the diversity of semantic of documents. Secondly, in visible and hidden units, the values vary continuously by each epoch in the training process. This leads to loss of semantic information of documents. One way to solve this problem is increasing the number of hidden units for RBM to accommodate more information. However, increasing the number of hidden units causes the model to increase the number of visible units. This leads to the problem encountered in the previous approach [4], which is insufficient computational resources.

Therefore, we propose Word Embedding Restricted Boltzmann Machine (WE-RBM) to take full advantage of WEM’s structure and supervised RBM. Beside input units, WE-RBM also uses prior knowledge obtained from the vector comparing process of WEM. Hence, the model would have more information to make up for the loss in the training process. Our WE-RBM model is expressed in Figure 3.

Assuming that WEM has been trained independently before, we put every document into WEM to generate the corresponding vector. The dimension of each vector is equal to the number of input units of RBM. After document vectors generation process, we make the training phase for the RBM model in a supervised setting. This phase helps the model be able to understand the vectors’ pattern generated by WEM.

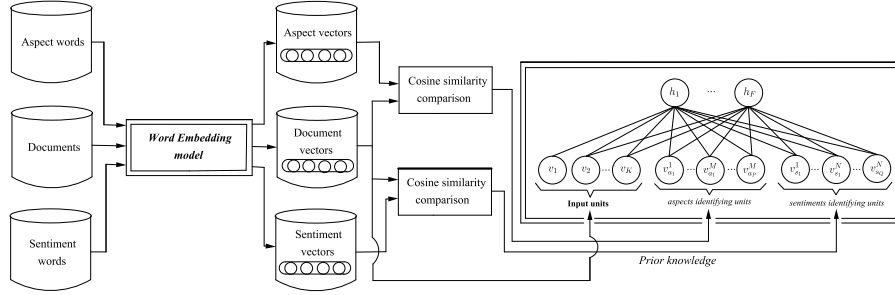


Fig. 3: WE-RBM model overview. Double boxed items are main components in WE-RBM model.

In the testing phase, each document is also converted into a vector by WEM. Unlike previous work, the aspect identification and sentiment classification task are firstly performed by WEM instead of RBM model. Particularly, WE-RBM computes the cosine similarity score (shown in equation 1) between the document vector and each of all aspect and sentiment vectors, which are also generated by WEM. The prior knowledge is determined by the highest similar pair of vectors.

For instance, a document  $d$  can be categorized into one of these aspects  $S = a_1, a_2, \dots, a_P$ . Let us call the vector form of document or aspect  $x \text{ vec}(x)$ . Prior knowledge would be labeled  $i$  if cosine similarity between  $\text{vec}(d)$  and  $\text{vec}(a_i)$  is the highest one. The model performs similarly with sentiment classification task. This prior knowledge can help improve the ability of the model to extract aspects and identify sentiments.

One characteristic of WEM is two vectors which represent two different words would have high cosine similarity if these two words have close meaning to each other [21, 22] (e.g. “strong” may have close meaning with “powerful”, whereas “strong” and “London” are more distant). Inspired by this, we use WEM to categorize aspect and classify sentiment before feeding the input vector into RBM model. Hence, each document can have its own prior knowledge before RBM’s classification. The final classification result is determined by both WEM and RBM. Therefore, we call our novel combination model is WE-RBM.

There are three main reasons we propose this novel WE-RBM model to solve the ABSA’s tasks. Firstly, when a document is converted into vector form, it can be compared with other documents by cosine similarity while semantic information is conserved. This is the advantage of word representation technique that WE-RBM use in the classification process. Secondly, WE-RBM can save computational resources and processing time. The model does not need to adapt input units for huge dictionary size thanks to fixed dimension of the input vector. Using prior knowledge instead of increasing hidden units, WE-RBM can handle big training set in reduced processing time. Last but not least, our WE-RBM model also has the capability of jointly modeling aspect and sentiment information together.

**Training and Testing** In the training process, Contrastive Divergence (CD), also called Approximate Gradient Descent, is the way that helps RBM learn the connection weights in the network. CD has two phases, which are Positive phase and Negative phase. In each phase, we compute the positive and negative value by the multiplication of visible and hidden units. Then, we update the connection weight based on the subtraction from positive value of negative value. Particularly, each epoch of CD can be expressed in five steps below:

**Step 1.** Update the states of the hidden units using the logistic activation rule described in equation 2. For the  $j$ -th hidden unit, compute its activation energy and set its state to 1 with corresponding probability.

$$P(h_j = 1|\mathbf{v}) = \text{sigm}(a_j + \sum_{i=1}^D \sum_{k=1}^K v_i^k W_{ij}^k) \quad (2)$$

**Step 2.** For each connection edge  $e_{ij}$ , get the value from positive phrase by equation 3.

$$\text{pos}(e_{ij}) = v_i h_j \quad (3)$$

**Step 3.** Reconstruct the visible units in a similar manner by using the logistic activation rule described in equation 4. For the  $i$ -th visible unit, compute its activation energy and set its state to 1 with corresponding probability. Then do **Step 1** to update the hidden units again.

$$P(v_i^k = 1|h) = \text{sigm}(b_i^k + \sum_{j=1}^F h_j W_{ij}^k) \quad (4)$$

**Step 4.** For each connection edge  $e_{ij}$ , get the value from negative phrase by equation 5.

$$\text{neg}(e_{ij}) = v_i h_j \quad (5)$$

**Step 5.** Update the connection weights  $W_{ij}$  by equation 6.

$$W_{ij} = W_{ij} + \text{lr}(\text{pos}(e_{ij}) - \text{neg}(e_{ij})) \quad (6)$$

where  $\text{sigm}(x) = \frac{1}{(1+e^{-x})}$  is the logistic function, and  $\text{lr}$  is a learning rate.

After  $m$  epochs of transfer between visible and hidden layers in a CD- $m$  run of the above steps, values of the hidden units reflect the relationship between the visible units in the model. The connection weight matrix between the two layers helps hidden layer generate visible units which include input units, aspect identifying units and also sentiment identifying units.

In the testing process, we convert all documents into vector form using WEM. Then, each document vector is compared with each aspect vector by cosine similarity score. The most similar pair of vectors gives the model prior knowledge about the aspect label of this document. Final aspect label of the document is determined based on the result of aspects identifying units after WE-RBM generation process.



## 4 Experimental results

In this section, we present two experiments to evaluate the performance of our model on the aspect identification and sentiment classification tasks.

### 4.1 Data

To evaluate our model performance, we used Restaurant Review Dataset [23]. This data is also used widely in previous work [23, 16, 24]. Data contain reviews about the restaurant with 1,644,923 tokens and 52,574 documents in total. Documents in this dataset are annotated with one or more labels from a gold standard label set  $S = \{Food, Staff, Ambience, Price, Anecdote, Miscellaneous\}$ .

### 4.2 Aspect Extraction

**Experimental Setup** Following the previous studies (Brody and Elhadad [16] and Zhao et al. [24]), reviews with less than 50 sentences are chosen. From that, we only use sentences with a single label for evaluation to avoid ambiguity. These sentences are selected from reviews with three major aspects chosen from the gold standard labels  $S' = \{Food, Staff, Ambience\}$ . After choosing suitable sentences from the data, we have 50303, 18437, 10018 sentences which labeled Food, Staff, Ambience. Then, we change the case of any document to lower case and remove stop words.

To convert each word in the sentence into vector form, we use Word Embedding technique [21] combined with Google pre-trained model<sup>1</sup>. This model has been trained on part of Google News dataset (about 100 billion words). It contains 300-dimensional vectors for 3 million words and phrases. Each sentence's representation is a vector which is a sum of vectors that represent for words appeared in that sentence [20].

We use 300 visible units in our WE-RBM as aspects identifying units, where units 1-100 capture *Food* aspect, units 101-200 capture *Staff* aspect and units 201-300 capture *Ambience* aspect. For sentiment classification, we also use 200 visible units, where units 1-100 capture positive information, units 101-200 represent negative information. Initially, these units are set to 0. After Gibbs sampling process, we create a sum for each group of 100 units to determine the aspect and sentiment polarity appeared in the document.

**Evaluation** To evaluate the model's performance, we use Precision, Recall and  $F_1$  scores for each aspect identification on Restaurant Review Dataset.

As a baseline, we implement *Prior knowledge only* method, which only uses cosine similarity of the document vector and aspect vector to identify aspect. For detail, we generate the vectors of "food", "staff" and "ambience" words using WEM. Then, we compute the cosine similarity between the vector of document

<sup>1</sup> <https://code.google.com/archive/p/word2vec/>

$d_i$  and each of those 3 aspect vectors. The document will be put into aspect  $a_i$  category if the cosine similarity between  $vec(d_i)$  and  $vec(a_i)$  is the highest one.

We also use SVM, a well-known machine learning technique [25], to compare with our WE-RBM model. We implement SVM by LIBSVM<sup>2</sup> which is an integrated tool for support vector classification. With linear kernel, SVM model is trained with feature vectors generated by bag-of-words model. Without prior knowledge, standard RBM using Word Embeddings is also re-implemented. This method processes the same Restaurant Review Dataset and identifies aspects for every document in this dataset under the same experimental conditions. Evaluation results for aspect identification are given in Table 1.

Table 1: Aspect identification results in terms of precision, recall, and  $F_1$  scores on the restaurant reviews dataset

Aspect	Method	Precision	Recall	$F_1$
Food	Prior knowledge only	88.09	87.60	87.84
	SVM	<b>94.10</b>	75.74	83.93
	RBM + Word Embeddings	74.58	14.79	24.68
	WE-RBM	84.04	<b>94.43</b>	<b>88.93</b>
Staff	Prior knowledge only	<b>95.96</b>	45.39	61.63
	SVM	90.25	<b>78.14</b>	<b>83.76</b>
	RBM + Word Embeddings	27.73	53.32	36.48
	WE-RBM	88.27	65.28	75.06
Ambience	Prior knowledge only	16.02	<b>86.93</b>	27.06
	SVM	29.15	82.70	43.11
	RBM + Word Embeddings	14.85	60.12	23.81
	WE-RBM	<b>69.22</b>	59.64	<b>64.07</b>

**Discussion** Considering the results from Table 1, we find that WE-RBM performs better than other methods. Specifically, it is evident that our WE-RBM model outperforms previous methods'  $F_1$  scores on *Food* and *Ambience* aspects. Compared with Prior knowledge only, the  $F_1$  scores improve by 1.09%, 13.43% and 37.01% respectively, for the *Food*, *Staff*, and *Ambience* aspects. This result proves that our WE-RBM model is not entirely based on Prior knowledge obtained from WEM to have the better performance. Inheriting RBM's ability in modeling latent topics and WEM's capability in identifying aspects, WE-RBM model can achieve higher Precision and Recall scores for the imbalanced dataset. Compared with RBM using Word Embeddings, the  $F_1$  scores yield relative improvements by 63.25%, 38.58%, and 40.26% respectively, on the same aspects. This result reveals that RBM model performs badly without the prior knowledge obtained from WEM.

<sup>2</sup> <https://www.csie.ntu.edu.tw/~cjlin/libsvm/>

Comparing with SVM performance, precision scores in food and staff domains of SVM are higher than WE-RBM's. But SVM can not address the imbalanced data problem, which results in reduced precision in *Ambience* domain. Moreover, WE-RBM is a joint model which has ability to identify aspects and classify sentiment polarities simultaneously, while SVM is a classification model and we have to train two separated models to solve these two tasks.

In our model's performance evaluation process, we can not re-implement SERBM due to insufficient computational resources. However, we compare WE-RBM's result with SERBM's result which was presented by Wang et al. [4]. With the same settings in aspect identification task on Restaurant Review Dataset, WE-RBM outperforms SERBM with improvement in  $F_1$  scores by 1.73% and 7.06% on *Food* and *Staff* aspects, respectively.

### 4.3 Sentiment Classification

Sentiment classification task is modified similar to aspect identification task in our WE-RBM model. We assign a sentiment score to every document in the Restaurant Review Dataset based on the output of WE-RBM's sentiment identifying units in the visible layer. Then we use SentiWordNet<sup>3</sup>, a famous lexical resource for opinion mining [26], and adapt SVM to compare the result with our WE-RBM model.

Following the previous study [4], we consult SentiWordNet to obtain a sentiment label for every word and aggregate these to judge the sentiment information of an entire review document in terms of the sum of word-specific scores. For SVM, we use the linear kernel to train the model and the other setting is the same as WE-RBM's. Table 2 shows the comparison between SentiWordNet, SVM and WE-RBM with Accuracy as the evaluation metric.

Table 2: Accuracy of SentiWordNet, SVM and WE-RBM on sentiment classification task

Method	Accuracy
SentiWordNet	73.36
SVM	78.26
WE-RBM	<b>79.79</b>

As we can observe in Table 2, the best sentiment classification accuracy result is 79.79% achieved by WE-RBM. Compared with two baselines, our WE-RBM yields a relative improvement in the overall accuracy by 6.43% over SentiWordNet and by 1.53% over SVM. Comparing the result of WE-RBM with SERBM's which was presented by Wang et al. [4], WE-RBM increases the classification accuracy by 0.99%.

<sup>3</sup> <http://sentiwordnet.isti.cnr.it>

The reason for better performance of WE-RBM compared with other methods is the combination of prior knowledge and generating process of RBM. Prior knowledge can be considered as the first classification phase of WE-RBM. This knowledge boosts WE-RBM classifier in both aspect and sentiment identification.

## 5 Conclusion and future work

In this paper, we have proposed Word Embedding Restricted Boltzmann Machines model to jointly identify aspect categories and classify sentiment polarities in the supervised setting. Our approach modifies standard RBM model and combine it with WEM, which not only helps reduce dimension of the input vectors but also gives the RBM model prior knowledge for more accurate classification. Our experimental results show that this model can outperform state-of-the-art models.

In the future, we plan to collect Vietnamese review dataset with diverse domains and combine our WE-RBM with stacked RBMs to form Deep Belief Networks. We hope that it will solve the aspect identification and sentiment classification tasks as well as RBM family methods.

## References

1. M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proceedings of KDD*, 2004, pp. 168–177.
2. S. Blair-goldensohn, T. Neylon, K. Hannan, G. A. Reis, R. McDonald, and J. Reynar, "Building a sentiment summarizer for local service reviews," in *In NLP in the Information Explosion Era*, 2008.
3. R. kumar V and K. Raghuveer, "Web user opinion analysis for product features extraction and opinion summarization," *International Journal of Web and Semantic Technology*, pp. 69–82, 2012.
4. L. Wang, K. Liu, Z. Cao, J. Zhao, and G. de Melo, "Sentiment-aspect extraction based on Restricted Boltzmann Machines," in *Proceedings of the 53rd Annual Meeting of the ACL*, 2015, pp. 616–625.
5. X. Ding, B. Liu, and P. S. Yu, "A holistic lexicon-based approach to opinion mining," in *Proceedings of the Conference on Web Search and Web Data mining*, 2008, pp. 231–240.
6. W. Wei and J. A. Gulla, "Sentiment learning on product reviews via sentiment ontology tree," in *Proceedings of 48th Annual Meeting of the ACL*, 2010, pp. 404–413.
7. Jiang, Long, M. Yu, M. Zhou, X. Liu, and T. Zhao, "Target-dependent twitter sentiment classification," in *Proceedings of the 49th annual meeting of the ACL*, 2011, pp. 151–160.
8. B. Erik and M.-F. Moens, "A machine learning approach to sentiment analysis in multilingual web texts," in *Information retrieval*, 2009, pp. 526–558.
9. W. Jin and H. H. Ho, "A novel lexicalized hmm-based learning framework for web opinion mining," in *Proceedings of international conference on machine Learning*, 2009, pp. 465–472.

10. R. Varghese and M. Jayasree, "Aspect based sentiment analysis using support vector machine classifier," in *Advances in Computing, Communications and Informatics (ICACCI)*, 2013, pp. 1581–1586.
11. C. Yejin and C. Cardie, "Hierarchical sequential learning for extracting opinions and their attributes," in *Proceedings of annual meeting of the ACL*, 2010, pp. 269–274.
12. N. Jakob and I. Gurevych, "Extracting opinion targets in a single-and cross-domain setting with conditional random fields," in *Proceedings of conference on empirical methods in NLP*, 2010, pp. 1035–1045.
13. B. Liu, *Sentiment Analysis and Opinion Mining*. Morgan and Claypool, 2012.
14. D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet Allocation," *Journal of Machine Learning Research* 3, pp. 993–1022, 2003.
15. I. Titov and R. McDonald, "A joint model of text and aspect ratings for sentiment summarization," in *Proceedings of annual meeting of the ACL*, 2008, pp. 308–316.
16. S. Brody and N. Elhadad, "An unsupervised aspect-sentiment model for online reviews," in *Proceedings of NAACL-HLT 2010*, 2010, pp. 804–812.
17. C. Lin and Y. He, "Joint sentiment/topic model for sentiment analysis," in *Proceedings of ACM - International conference on Information and Knowledge management*, 2009, pp. 375–384.
18. Q. Mei, X. Ling, M. Wondra, H. Su, and C. Zhai, "Topic sentiment mixture: Modeling facets and opinions in weblogs," in *Proceedings of International conference on World Wide Web*, 2007, pp. 171–180.
19. A. Fischer and C. Igel, "Training Restricted Boltzmann Machines: An Introduction," in *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, 2012, pp. 14–36.
20. Q. V. Le and T. Mikolov, "Distributed representations of sentences and documents," *CoRR*, vol. abs/1405.4053, pp. 1188–1196, 2014.
21. R. Řehůřek and P. Sojka, "Software Framework for Topic Modelling with Large Corpora," in *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. ELRA, 2010, pp. 45–50.
22. T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in Neural Information Processing Systems 26*. Curran Associates, Inc., 2013, pp. 3111–3119.
23. G. Ganu, N. Elhadad, and A. Marian, "Beyond the stars: Improving rating predictions using review text content," in *Proceedings of WebDB 2009*, 2009, pp. 1–6.
24. W. X. Zhao, J. Jiang, H. Yan, and X. Li, "Jointly modeling aspects and opinions with a MaxEnt-LDA hybrid," in *Proceedings of EMNLP 2010*, 2010, pp. 56–65.
25. C. Chih-Chung and L. Chih-Jen, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011, software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
26. S. Baccianella, A. Esuli, and F. Sebastiani, "Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining," in *Proceedings of the Seventh Conference on International Language Resources and Evaluation (LREC'10)*, 2010, pp. 2200–2204.