Semi-supervised Sentiment Classification Based on Auxiliary Task Learning

Huan Liu, Jingjing Wang, Shoushan Li(✉), Junhui Li, and Guodong Zhou

Natural Language Processing Lab, School of Computer Science and Technology, Soochow University, Suzhou, China
hliu0909@stu.suda.edu.cn, djingwang@gmail.com, {lishoushan,lijunhui,gdzhou}@suda.edu.cn

Abstract. Sentiment classification is an important task in the community of Nature Language Processing. This task aims to determine the sentiment category towards a piece of text. One challenging problem of this task is that it is difficult to obtain a large number of labeled samples. Therefore, a large number of studies are focused on semi-supervised learning, i.e., learning information from unlabeled samples. However, one disadvantage of the previous methods is that the unlabeled samples and the labeled samples are studied in different models, and there is no interaction between them. Based on this, this paper tackles the problem by proposing a semi-supervised sentiment classification based on auxiliary task learning, namely Aux-LSTM, which is used to assist learning the sentiment classification task with a small amount of human-annotated samples by training auto-annotated samples. Specifically, the two tasks are allowed to share the auxiliary LSTM layer, and the auxiliary expression obtained by the auxiliary LSTM layer is used to assist the main task. Empirical studies demonstrate that the proposed method can effectively improve the experimental performance.

Keywords: Sentiment classification · Auxiliary task · Auto-annotation samples

1 Introduction

With the development of the social media, people are accustomed to commenting on characters, events and products on the internet to express their opinion and sentiment. Sentiment analysis is a hot research direction that is produced under such a background, and text sentiment classification is the basic task in sentiment analysis. Specifically, the task of text sentiment classification is to determine the sentiment orientation of a text, i.e., *positive* and *negative*. For example, the text "*This book is simply boring!*" is considered as a *negative* sentiment. There are a large number of product comments in the electronic commerce platform. Correctly identifying the sentiment of these comments helps to understand the evaluation of the products, thereby improving the product quality and providing better service to customers. From the perspective of customers, they can judge the quality of products by analyzing the sentiment of comments, so as to make correct choices.

M. Zhang et al. (Eds.): NLPCC 2018, LNAI 11109, pp. 372–382, 2018.
https://doi.org/10.1007/978-3-319-99501-4_33

Early research on sentiment classification mainly focus on supervised learning using only labeled samples [1, 2]. However, supervised learning requires a large number of labeled samples, studies in recent years have used semi-supervised learning method to reduce the dependence on labeled samples. For example, collaborative training (Co-training) [3], label propagation (LP) [4] and deep learning [5] are widely used in semi-supervised sentiment classification task. This paper mainly focuses on the method of semi-supervised sentiment classification.

At present, several studies have verified the effectiveness of the semi-supervised sentiment classification method, i.e., the use of unlabeled samples can improve the performance of sentiment classification. However, these existing methods have their own advantages and disadvantages, so it is difficult to determine which algorithm is suitable for which domain of sentiment classification. For example, the co-training algorithm can achieve good performance in the domain of Book and Kitchen, while the LP algorithm has better performance in the domain of DVD and Electronic [6]. Therefore, the semi-supervised sentiment classification method of integrated learning is also produced, which can improve the performance of sentiment classification through multiple semi-supervised learning methods. The above methods are aimed at reducing the error of annotating unlabeled samples in semi-supervised learning.

However, the unlabeled samples and labeled samples in the above semi-supervised learning algorithm are usually studied in two different models, ignoring the loss correlation information between models. In order to further study the link between the labeled samples and unlabeled samples information, and reduce the error of annotating unlabeled samples, we propose a semi-supervised sentiment classification method based on auxiliary task learning. The method firstly annotates unlabeled samples automatically, so as to obtain auto-annotated samples. Then, two sentiment classification tasks, the main task and the auxiliary task, are designed respectively according to the human-annotated samples and the auto-annotated samples. The main task obtains the auxiliary representation through the auxiliary LSTM layer, which is shared with the auxiliary task, and adds this auxiliary representation to the main task to assist the main task to complete the sentiment classification. The experimental results show that the proposed method in this paper can effectively improve the semi-supervised sentiment classification performance by utilizing the information of unlabeled samples.

The rest of our paper is structured as follows. Section 2 briefly discusses the related work. Section 3 gives the overall framework. Section 4 describes the algorithm of obtaining auto-annotated samples. Section 5 describes the semi-supervised sentiment classification based on auxiliary task learning in detail. Section 6 gives experimental settings and experimental results. Finally, the last section is the conclusion of this paper.

2 Related Work

Early sentiment classification research mainly focus on supervised learning. Ye et al. [1] compare three supervised machine learning algorithms of Naïve Bayes, SVM and the character based *n*-gram model for sentiment classification of the reviews on travel blogs for seven popular travel destinations in the US and Europe. Pang et al. [2]

introduce a variety of classification methods into the task of sentiment classification and achieves good classification results.

Since supervised learning requires a large number of labeled samples, the semi-supervised learning method has gradually attracted the attention of researchers. Wan [3] takes two different languages (English and Chinese) as two different views and adopts co-training method to semi-supervised sentiment classification. Zhu and Ghagramni [4] propose a graph-based semi-supervised learning method, namely label propagation (LP). The basic idea is to use the relationship between the samples to establish a relational complete graph model. In the complete graph, the nodes include labeled and unlabeled samples, and the edges represent the similarity of the two nodes. The labels of the nodes are transmitted to other nodes according to similarity. Xia et al. [7] make use of the original and antonymous views in pairs, in the training, bootstrapping and testing process, all based on a joint observation of two views. Sharma et al. [8] propose a semi-supervised sentiment classification method that uses sentiment bearing word embedding form to generate continuous ranking of adjectives with common semantic meaning. Yu and Jiang [9] borrow the idea from Structural Correspondence Learning and use two auxiliary tasks to help induce a sentence embedding that supposedly words well across domains for sentiment classification.

Different from traditional semi-supervised learning method, this paper proposes a semi-supervised sentiment classification method based on auxiliary task learning, which is constructing the joint loss function of the main task and the auxiliary task. It eliminates the need to add auto-annotated samples to human-annotated samples for modeling, thereby reducing the error of annotating unlabeled samples.

3 Overall Framework

Figure 1 shows the overall framework of semi-supervised sentiment classification based on auxiliary task learning. The basic idea is making human-annotated samples, i.e., labeled samples, and auto-annotated samples to learn from each other to assist completing the sentiment classification. Specifically, two sentiment classification tasks are designed, i.e., a main task and an auxiliary task. The main task implements the sentiment classification of human-annotated samples. The auxiliary task implements the sentiment classification of auto-annotated samples. The two tasks share the auxiliary LSTM layer, i.e., the auxiliary task completes the auxiliary sentiment classification through the auxiliary LSTM layer, and the main task completes the main sentiment classification with the auxiliary expression obtained by the auxiliary LSTM layer. Finally, joint learning the loss function of the two task to improve the performance of the main task.

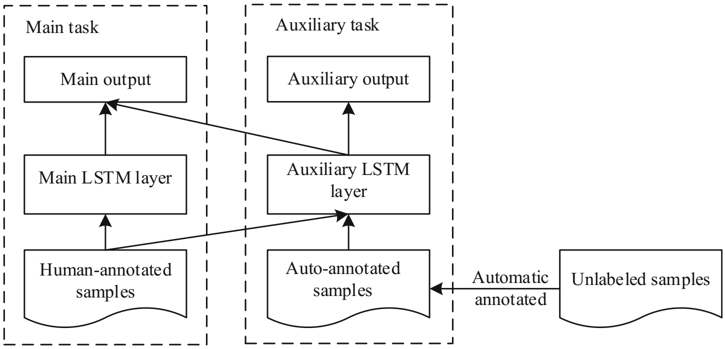

Fig. 1. The overall framework

4 Automatic Labeling Method for Unlabeled Samples

In order to complete the experiment of the semi-supervised sentiment classification method proposed in this paper, we need to obtain the label of the unlabeled samples. Firstly, we use the information gain (IG) algorithm [10–12] to extract 1000 *positive* feature words from *positive* labeled samples and 1000 *negative* feature words from *negative* labeled samples. The extraction method of *positive* feature words is to calculate the IG value of each word appearing in the *positive* samples, sort the IG values in descending order, and take the words corresponding to the first 1000 IG values as the *positive* feature words. The extraction method of *negative* feature words is the same as the *positive* feature words. Then, the unlabeled samples are divided into *positive* and *negative* categories according to the number of *positive* and *negative* feature words included in each sample in the unlabeled sample, wherein only the occurrence or non-occurrence of the feature words is considered, and the frequency is not considered. The specific algorithm is shown in Fig. 2:

Input:

Positive feature set P;
Negative feature set N;
Unlabeled samples U;

Output:

Auto-annotated samples T

Procedure:

Loop until $U = \varnothing$

1) Calculate the number of *positive* words C_P contained in the sample according to the set P
2) Calculate the number of *negative* words C_N contained in the sample according to the set N
3) If $C_P > C_N$, label the sample as *positive*
   If $C_P < C_N$, label the sample as *negative*
   If $C_P = C_N$, label the sample randomly
4) Add the new auto-annotated sample into T

Fig. 2. The algorithm of obtaining auto-annotated samples

5 Semi-supervised Sentiment Classification Based on Auxiliary Task Learning

This section mainly introduces semi-supervised sentiment classification method based on auxiliary task learning. First, we introduce the basic LSTM neural network. Second, we propose a method of the sentiment classification of human-annotated samples sharing the auxiliary LSTM layer with the sentiment classification of auto-annotated samples, so as to make full use of the information of the unlabeled samples.

5.1 LSTM Model for Sentiment Classification

Long Short-Term Memory Network (LSTM) is a special kind of Recurrent Neural Network (RNN) and it aims to learn long-dependency correlations in a sequence [13]. We adopt the standard LSTM layer used by Graves [14].

First, the one-hot feature representation of the text T is used as an input to the LSTM layer and a new representation h is obtained. The formula is as follows:

$$h = LSTM(T) \tag{1}$$

Then, the output of the LSTM layer is propagated to the fully connected layer, and the output of the fully connected layer h^* is obtained by weighting the activation function, i.e.,

$$h^* = dense(h) = \phi(\theta^T h + b) \tag{2}$$

where ϕ is a non-linear activation function, and "Relu" is used as an activation function [15]. "Relu" will cause the output of some neurons in the network to be 0, which reduces the dependence between parameters and is closer to the biological activation model, alleviating the occurrence of overfitting. θ^T is the weight matrix and b is the bias.

In order to reduce the model complexity and prevent the network from overfitting the training samples, we add the dropout layer after the fully connected layer. The dropout layer randomly ignores some hidden units in the network during training. This paper uses the dropout layer as a hidden layer in the network:

$$h^d = h^* \cdot D(p^*) \tag{3}$$

where D denotes the dropout operation and p^* denotes the dropout probability. h^d is the output of h^* after the dropout layer operation.

Finally, using the softmax layer to complete the classification task, the predicted probability is obtained by the following formula:

$$p = softmax(W^d h^d + b^d) \tag{4}$$

where W^d and b^d are the parameters for the softmax layer. p is the conditional probability distribution over the two categories of sentiment, i.e., *positive* and *negative*.

5.2 Aux-LSTM Model for Semi-supervised Sentiment Classification

Figure 3 shows the overall architecture of the semi-supervised sentiment classification method based on auxiliary LSTM (Aux-LSTM), which mainly includes one main task and one auxiliary task.

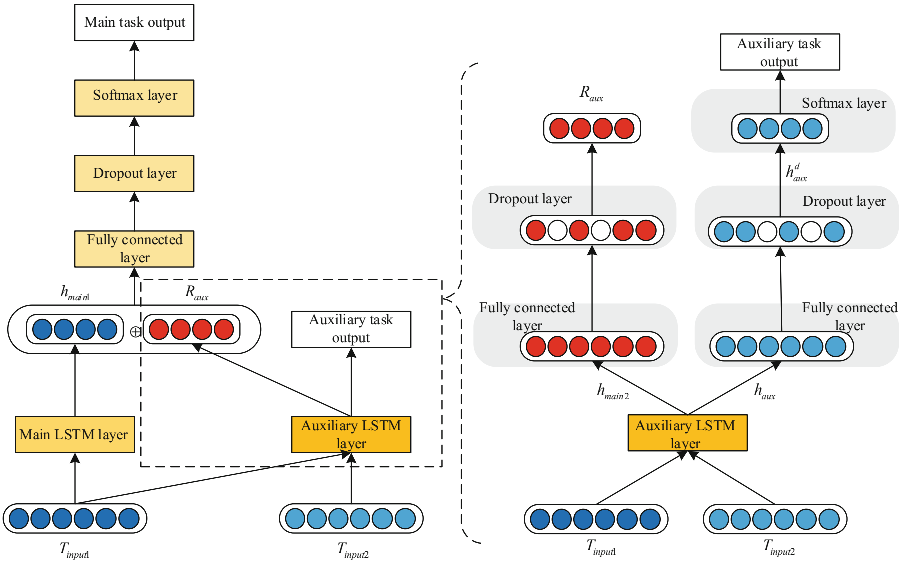

Fig. 3. Overall architecture of Aux-LSTM for semi-supervised sentiment classification

- **The main task-using human-annotated samples**

This part describes the main task of the semi-supervised sentiment classification method, consisting of the main LSTM layer and the auxiliary LSTM layer:

$$h_{main1} = LSTM_{main}(T_{input1}) \tag{5}$$

$$h_{main2} = LSTM_{aux}(T_{input1}) \tag{6}$$

where T_{input1} represents the one-hot feature representation of the human-annotated samples. h_{main1} and h_{main2} represent the output through the main LSTM layer ($LSTM_{mian}$) and the auxiliary LSTM layer ($LSTM_{aux}$) respectively.

Then, we feed h_{main2} to the fully connected layer and get an auxiliary representation R_{aux} through a dropout layer:

$$R_{aux} = dense(h_{main2}) \cdot D(p^*) \tag{7}$$

We can obtain a novel representation after concatenating above two representation h_{main1} and R_{aux} and use them as the input of a fully connected layer followed by a dropout layer in the main task:

$$h^{d}_{main} = dense(h_{main1} \oplus R_{aux}) \cdot D(p^{*}) \tag{8}$$

where $\oplus$ denotes the concatenate operator.

Finally, softmax layer is used to complete the classification, refer to 5.1 for details.

- **The auxiliary task-using auto-annotated samples**

This part describes the auxiliary task of the semi-supervised sentiment classification method. The auxiliary LSTM layer, which is the LSTM layer shared by the main task and the auxiliary task, has the same input sequence and weight as the auxiliary LSTM layer in the main task:

$$h_{aux} = LSTM_{aux}(T_{input2}) \tag{9}$$

where T_{input2} denotes the one-hot feature representation of the auto-annotated samples.

Then, we use the output of the auxiliary LSTM layer as the input to the hidden layer:

$$h^{d}_{aux} = dense(h_{aux}) \cdot D(p^{*}) \tag{10}$$

Finally, softmax layer is used to complete the classification, refer to 5.1 for details.

- **Joint learning-using both human-annotated and auto-annotated samples**

In order to better learn the parameters of the auxiliary LSTM layer in the model, we weighted the loss function of the main task and the auxiliary task to obtain the joint learning loss function, i.e.,

$$loss = \lambda(loss_{main}) + (1 - \lambda)(loss_{aux}) \tag{11}$$

where λ denotes the weight parameter, here we set 0.75 to reduce the noise of the auxiliary task. $loss_{main}$ is the loss function of the main task, while $loss_{aux}$ is the loss function of the auxiliary task. We take Adam [16] as our optimizing algorithm.

6 Experiments

6.1 Experimental Settings

In this paper, we use the corpus of Amazon product reviews, which is annotated by Blitzer et al. [17]. The corpus consist of four domains: Book, DVD, Electronics, and Kitchen. In the experiment of each domain, we select 100 instances as labeled data for training and 400 instance are used as test. The task of the experiment is to determine whether the sentiment of a text is *positive* or *negative*. According to the number of unlabeled samples, we make four sets of experiments. The number of training samples for each set of auxiliary tasks, i.e., the number of auto-annotated samples, is 750, 950, 1200, and 1450 respectively. The test samples for each set of experiments in auxiliary task is the same as the main task.

The classification feature used in the experiment is the one-hot representation of text. Specifically, we first construct a dictionary in descending order of occurrence frequency of word features in all corpus, and then use the subscripts of the word in the dictionary, thereby constructing the feature vectors of the samples. We use the LSTM neural network as the basic classification algorithm. The specific parameter settings of the LSTM neural network model are shown in Table 1.

Table 1. Parameters setting in LSTM

Parameter description	Value
Dimension of the LSTM layer output	128
Dimension of the full-connected layer output	64
Dropout probability	0.5
Epochs of iteration	30

We employ accuracy to measure the performance of the classification. It gives an average degree of the similarity between the predicted and ground truth label sets of all test samples, i.e.,

$$Accuracy = \frac{1}{m}\sum_{i=1}^{m} 1_{y_i = y_i^{'}} \tag{12}$$

where m is the number of all test samples, y_i is the true label and $y_i^{'}$ is the estimated label.

6.2 Experiments

For thorough comparison, we implement the following approaches to semi-supervised sentiment classification:

- **ME:** We employ the maximum entropy classifier in the MALLET Machine Learning Toolkit[1]. All parameters of the algorithm are set to default values. Here we only use the human-annotated samples to train the classification model.
- **Co-training:** The idea of the Co-training algorithm is to train multiple classifiers with multiple independent views, and then iteratively expands labeled samples and retrains them using that new labeled samples. In the implementation, we use each feature subspace as a representation view of text, and multiple feature subspaces correspond to different views.

[1] http://mallet.cs.umass.edu/.

- **LP:** LP algorithm uses the relationship between the samples to establish a relational complete graph model. In the complete graph, the nodes include labeled and unlabeled samples, and the edges represent the similarity of the two nodes. The labels of the nodes are transmitted to other nodes according to similarity.
- **Semi-stacking:** Li et al. [18] integrate two or more semi-supervised learning algorithms from an ensemble learning perspective. Specifically, they apply *meta*-learning to predict the unlabeled samples and proposed *N*-fold cross validation to guarantee a suitable size of the data for training the *meta*-classifiers.
  LSTM: We use the standard LSTM model, which includes a LSTM layer, a fully connected layer, and a dropout layer. The method used here for unlabeled samples is to train the one-hot feature with the labeled and unlabeled samples.
- **Aux-LSTM:** The method of the auxiliary LSTM described in Sect. 5.

In this paper, four sets of experiments are conducted on sentiment classification based on different numbers of auto-annotated samples. Figure 4 shows the experimental results which the number of the auto-annotated samples is 2900.

From Fig. 4, it can be seen that the results of sentiment classification using auto-annotated samples are significantly better than those using only human-annotated samples, i.e., ME. The method of Co-training has significantly improved in the domains of DVD, Book and Kitchen, but has not improved in the Electronic. However, the method of LP has slightly improved in the domains of DVD, Electronic and Kitchen, but there is almost no improvement in the Book. Semi-stacking combines the advantages of the Co-training and LP algorithms, and the accuracy in the four domains is significantly improved. The results of our method (Aux-LSTM) in four domains are clearly superior to those using only human-annotated samples for sentiment classification. Although performance similar to the Semi-stacking method was achieved in the domain of Kitchen (accuracy is 0.2% lower), performance in the DVD, Book, and Electronic was significantly higher than other semi-supervised methods. For example, in the domains of DVD, Book, and Electronic, our method has improved 4%, 2.3% and 2.8%

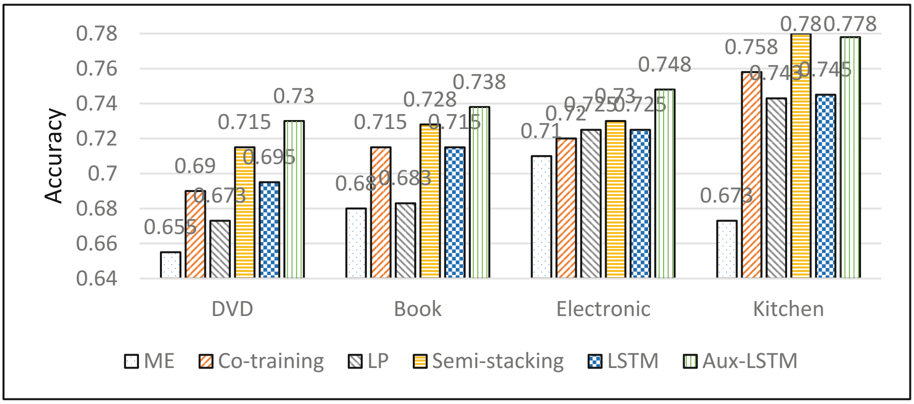

Fig. 4. Performances of different approaches to semi-supervised sentiment classification in four domains (The number of the auto-annotated samples is 2900)

respectively compared with Co-training method. This result fully shows that the Aux-LSTM model can effectively reduce the impact of incorrect auto-annotated samples on the semi-supervised sentiment classification task, and can better improve classification performance than other traditional semi-supervised sentiment classification methods.

7 Conclusion

This paper proposes a semi-supervised sentiment classification method based on auxiliary task learning. The method first annotates the unlabeled samples automatically with IG algorithm to obtain the auto-annotated samples. Then, it assists in sentiment classification of the human-annotated samples (main task) through the sentiment classification of the auto-annotated samples (auxiliary task). Finally, joint learning the loss function of the two task to improve the performance of the main task. The experimental results show that the semi-supervised sentiment classification method proposed in this paper can make full use of unlabeled samples to improve the performance of sentiment classification, and is superior to the current mainstream semi-supervised sentiment classification methods.

Acknowledgments. This research work has been partially supported by two NSFC grants, No. 61672366 and No. 61331011.

References

1. Ye, Q., Zhang, Z., Law, R.: Sentiment classification of online reviews to travel destinations by supervised machine learning approaches. Expert Syst. Appl. **36**(3), 6527–6535 (2009)
2. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up? Sentiment classification using machine learning techniques. In: Proceedings of EMNLP, Philadelphia, pp. 79–86 (2002)
3. Wan, X.: Co-training for cross-lingual sentiment classification. In: Proceedings of ACL-IJCNLP, Singapore, pp. 235–243 (2009)
4. Zhu, X., Ghahramani, Z.: Learning from Labeled and Unlabeled Data with Label Propagation. CMU CALD Technical report, CMU-CALD-02-107
5. Zhou, S., Chen, Q., Wang, X.: Active deep networks for semi-supervised sentiment classification. In: Proceedings of COLING, Beijing, pp. 1515–1523 (2010)
6. Li, S., Xue, Y., Wang, Z., et al.: Active learning for cross-domain sentiment classification. In: Proceedings of IJCAI, Beijing, pp. 2127–2133 (2013)
7. Xia, R., Wang, C., Dai, X., et al.: Co-training for semi-supervised sentiment classification based on dual-view bags-of-words representation. In: Proceedings of ACL-IJCNLP, Beijing, pp. 1054–1063 (2015)
8. Sharma, R., Somani, A., Kumar, L., et al.: Sentiment intensity ranking among adjectives using sentiment bearing word embeddings. In: Proceddings of EMNLP, pp. 547–552 (2017)
9. Yu, J., Jiang, J.: Learning sentence embeddings with auxiliary tasks for cross-domain sentiment classification. In: Proceddings of EMNLP, pp. 236–246 (2016)
10. Shi, H., Jia, D., Miao, P.: Improved information gain text feature selection algorithm based on word frequency information. J. Comput. Appl. **34**, 3279–3282 (2014)
11. Xu, J., Jiang, H.: An improved information gain feature selection algorithm for SVM text classifier. In: Proceddings of IEEE, pp. 273–276 (2015)

12. Zhang, H., Ren, Y.G., Yang, X.: Research on text feature selection algorithm based on information gain and feature relation tree. In: Proceddings of IEEE, pp. 446–449 (2014)
13. Hochreiter, K., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
14. Graves, A.: Generating sequences with recurrent neural networks. Computer Science (2013)
15. Glorot, X., Bordes, A., Bengio, Y.: Deep sparse rectifier neural networks. In: JMLR W&CP, vol. 15, pp. 315–323 (2012)
16. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. Computer Science (2014)
17. Blitzer, J., Dredze, M., Pereira, F.: Biographies, bollywood, boom-boxes and blenders: domain adaptation for sentiment classification. In: Proceedings of ACL, Prague, pp. 440–447 (2007)
18. Li, S., Huang, L., Wang, J., et al.: Semi-stacking for semi-supervised sentiment classification. In: Proceedings of ACL-IJCNLP, Beijing, pp. 27–31 (2015)