International Journal of Signal Processing, Image Processing and Pattern Recognition
Vol.9, No.9, (2016), pp.205-214
http://dx.doi.org/10.14257/ijsip.2016.9.9.19

Visual Sentiment Analysis with Network in Network

Zuhe Li[1,2], Yangyu Fan[1] and Fengqin Wang[2]

[1]*School of Electronics and Information, Northwestern Polytechnical University, Xi'an 710072, China*
[2]*School of Computer and Communication Engineering,Zhengzhou University of Light Industry, Zhengzhou 450002, China*
zuheli@126.com

Abstract

In modern society, visual content like images and videos is increasingly becoming a new form of media to express users' opinions on the Internet. As a complement to textual sentiment analysis, visual sentiment analysis intends to provide more robust information for data analytics by extracting emotion and sentiment toward topics and events from images and videos. Inspired by recent works that applied deep convolutional neural networks (CNN) to this challenging problem, we proposed a framework for image sentiment analysis with a novel deep neural network called Network in Network (NIN) which intends to improve the discriminability for local patches within receptive fields. We trained our network on a dataset consisting of nearly half a million Flickr images and minimized the effect of noisy training data by fine-tuning the network in a progressive manner. Extensive experiments conducted on manually labeled Twitter images show that the proposed architecture performs better in visual sentiment analysis than conventional CNN and other traditional algorithms.

Keywords: *visual sentiment; deep learning; convolutional neural network; network in network*

1. Introduction

With the pervasion of images and videos on social media, visual sentiment analysis has attracted more and more attention. Understanding the strong emotional semantics in images and videos that could influence the audience will enable broad applications in many areas such as education, entertainment and advertisement [1-2]. Motivated by existing progress in text-based sentiment analysis, many researchers have started to extract sentiment information from visual content. However, this is a more challenging work compared to other problems in computer vision because it is associated with abstract human emotion and affection [3-4].

Though much promising progress has been achieved in textual sentiment analysis [1-5], research on visual content sentiment analysis is still limited. Some researchers have attempted to introduce commonly used techniques in computer vision to this area by mapping low-level visual features to sentiment or affection directly [6-7]. Nevertheless, "affective gap" between low-level features and affection has become a big problem that can't be solved by these methods [3]. Thus Siersdorfer *et. al.,* [8] proposed a scheme to predict visual sentiment with bag-of-visual words representation and color distribution. Borth *et. al.,* [3-4] and Yuan *et .al.,* [9] employed mid-level features like entities or attributes for visual sentiment analysis to overcome the problem of "affective gap".

So far, the most representative researches in visual sentiment analysis have been conducted by Columbia University's digital video and multimedia lab [2-4]. This team has constructed a Visual Sentiment Ontology (VSO) including more than 1,200 Adjective Noun Pairs (ANP) corresponding to different emotions. It can be used as mid-level

ISSN: 2005-4254 IJSIP

features to bridge the affective gap [3]. They also crawled images from Flickr using these ANPs and trained visual concept detectors to detect the responses of 1,200 ANPs in an image, called SentiBank [4]. For example, Figure 1, gives some images in this dataset corresponding to four ANPs which convey strong visual sentiment information. In the age of big data, deep learning architectures like CNNs have demonstrated excellent performance on several tasks in computer vision [10-13].Coincidentally, the large dataset built by Columbia University includes about half a million images and provides enough training data for deep learning algorithms. Thus it is possible to introduce the hot technologies of deep learning to visual sentiment analysis.

Figure 1. Sample Images from The Flickr Dataset for Sentiment Analysis

Some researchers have already begun to apply deep convolutional neural networks to visual content-based sentiment analysis. Chen *et. al.*, [4] proposed a deep CNN framework for visual sentiment concept classification and achieved great improvement on both annotation accuracy and retrieval performance. Similarly, You *et. al.*, [14] also utilized a CNN to analyze image sentiment and got unexpected results. These successes therefore indicated the feasibility of applying deep learning algorithms to visual sentiment analysis. Inspired by these works, we try to solve this challenging problem with a newly proposed deep network structure NIN [15] which achieves state-of-the-art classification performances on benchmark datasets. NIN is a special kind of CNN in which the generalized linear model (GLM) is replaced with a micro network structure like multilayer perceptron (MLP) [15].

In this paper, we present a deep NIN architecture with two MLP convolutional layers and several fully connected layers for visual sentiment analysis. We further fine-tune the neural network using a progressive training strategy since the dataset is weakly labeled by machine and there exist noisy images in the training subset. In the rest of the paper, we first introduce the NIN in Section 2 and then describe the details of the proposed framework and training process in Section 3. Then we present the experiments and results in detail in Section 4. We finally make a conclusion for this paper in Section 5.

2. Network in Network

Conventional CNNs are hierarchical networks with alternatively stacked convolutional layers and spatial subsampling layers [16]. In convolutional layers, feature maps are obtained by taking inner product of the linear filter and corresponding receptive fields followed by nonlinear activation functions such as sigmoid, tanh and rectifier. Formally, a linear model is a function $f : R^D \rightarrow R^K$, where D is the size of input vectors and K is the

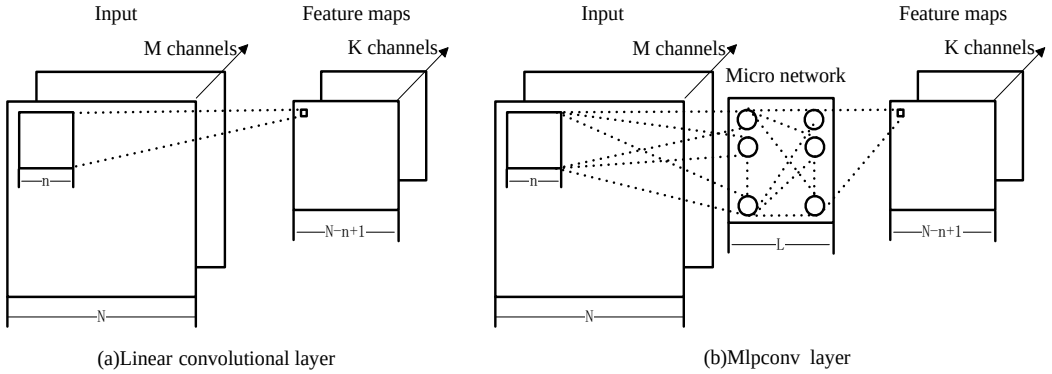
size of output vectors. Taking the sigmoid function as an example, feature maps can be obtained as follows:

$$a_{i,j} = f(z_{i,j}) = f(Wx_{i,j} + b) \tag{1}$$

Where $x_{i,j}$ is the input patch at location (i,j),W is the weight matrix, b is the bias vector and $f()$ is the activation function. The sigmoid function is expressed as:

$$f(z) = \frac{1}{1+\exp(-z)} \tag{2}$$

The convolutional filter in CNN is a generalized linear model with a low level of abstraction. Here abstraction indicates the invariance of features to local changes of the input [17]. In NIN, the GLM is replaced with a micro network structure to enhance the abstraction level of the local model [15]. The micro network structure is a general nonlinear function approximator like multilayer perceptron. Figure 2, shows the comparison of a linear convolutional layer in traditional CNNs and a convolutional layer with multilayer perceptron (*i.e.*, mlpconv layer [15]) in NIN.



(a)Linear convolutional layer (b)Mlpconv layer

Figure 2. Comparison of a Linear Convolution Layer and an mlpconv Layer

As shown in Figure 2, an mlpconv layer maps a local receptive field to output features in a different manner from that of a linear convolutional layer. An MLP with multiple fully connected layers and nonlinear activation functions is adopted to implement this mapping. An MLP is a function $f: R^{n\times n\times M} \to R^K$ where $n\times n\times M$ is the size of the input vector $x_{i,j}$ and K is the size of the output vector $a_{i,j}$. Taking an MLP one hidden layer for example,

$$a_{i,j} = f(W^{(2)} f(W^{(1)} x_{i,j} + b^{(1)}) + b^{(2)}) \tag{3}$$

with bias vectors $b^{(1)}$, $b^{(2)}$ and weight matrices $W^{(1)}$, $W^{(2)}$.The calculation performed by an mlpconv layer with multiple hidden layers is shown as follows:

$$\begin{aligned} a_{i,j}^{(2)} &= f(z_{i,j}^{(2)}) = f(W^{(1)} x_{i,j} + b^{(1)}) \\ &\vdots \\ a_{i,j}^{(l)} &= f(z_{i,j}^{(l)}) = f(W^{(l-1)} a_{i,j}^{(l-1)} + b^{(l-1)}) \\ &\vdots \end{aligned} \tag{4}$$

where l is the index of layers in the multilayer perceptron.

For convolution, MLPs are shared among all local receptive fields and slided over the input in the same way as a CNN to obtain the feature maps. Then the features are fed into the next layer and the stacking of multiple mlpconv layers forms a deep NIN.

3. Overall Structure and Progressive Fine-Tuning

Here we describe the overall structure of the proposed framework for visual sentiment analysis and the progressive fine-tuning strategy in the training process. The architecture of the NIN we employ is shown in Figure 3.

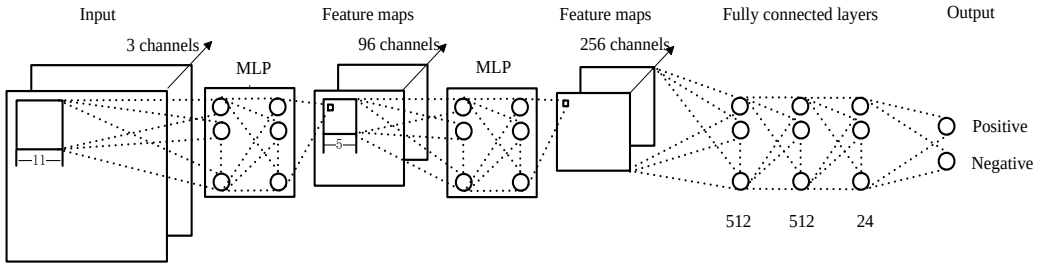


Figure 3. Network in Network for Visual Sentiment Analysis

To make a comparison with the CNN presented in [14] for visual sentiment analysis, we propose a similar framework by replacing the GLMs of a traditional CNN with MLPs. All images are first resized to 256×256 without considering the aspect ratio and 227×227 patches are extracted by cropping the middle part of the resized image. The normalized images are then fed into two mlpconv layers. In the first mlpconv layer, the input images are filtered with 96 kernels of size 11×11×3 and with a stride of 4 pixels. The second mlpconv layer has 256 kernels of size 5×5×96 with a stride of 2 pixels. Each mlpconv layer is followed by a max-pooling layer with pooling region size of 3×3 and a stride of 2 pixels, which is not shown in Figure 3 because of space limitation. In each mlpconv layer, the number of layers of an MLP is set to 2. The training images are grabbed from web according to the Plutchik's wheel of emotions [18] which consists of 24 positive and negative emotions. So we constrain the number of the second to last layer's nodes to 24 in order to model the mapping from 24 emotions to final classification results.

In the last layer, we use the logistic regression model to predict the value of the visual sentiment $y^{(i)}$ for the feature vector $x^{(i)}$ of the i-th example. We assume that the labels of visual sentiment are binary: $y^{(i)} \in \{0,1\}$, where 1 corresponds to "positive" and 0 corresponds to "negative". In logistic regression, a hypothesis is designed to denote the probability that an example belongs to the "1" class:

$$h_w(x) = p(y=1|x) = \frac{\exp(w^T x)}{1+\exp(w^T x)} \tag{5}$$

where w is the parameters of the logistic regression model. Then the probability that an example belongs to the "0" class is:

$$p(y=0|x) = 1\text{-}p(y=1|x) = \frac{1}{1+\exp(w^T x)} \tag{6}$$

And the model is trained to minimize the cost function:

$$J(w) = -\sum_i (y^{(i)} \log(h_w(x^{(i)})) + (1-y^{(i)}) \log(1-h_w(x^{(i)}))) \tag{7}$$

where $x^{(i)}$ is the feature vector for the i-th instance and $y^{(i)}$ is the label.

The images for visual sentiment analysis collected by digital video and multimedia lab of Columbia University were automatically grabbed from the Internet and labeled by

machine [3].Thus the noise in the training subset may make the network get stuck at a bad local optimum in the training process. To overcome this problem, You *et. al.*, [11] have presented a progressive strategy called progressive CNN (PCNN) in order to fine-tune the network for visual sentiment analysis. Similarly, we also employ a method to fine-tune the NIN by sampling the training subset progressively. The overall flow of the progressive NIN (PNIN) is shown in Figure 4.

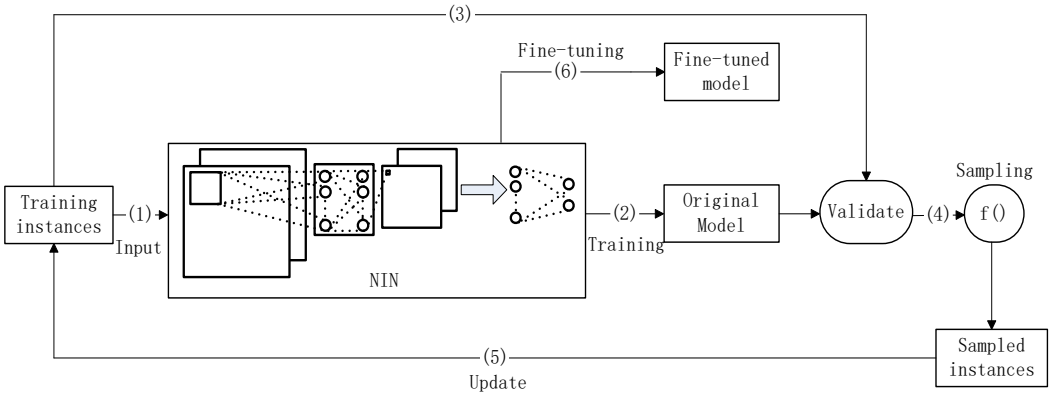


Figure 4. Overall Flow of the Progressive Fine-Tuning Strategy

The NIN is first trained with noisy training images directly. Then the trained model is used to validate the training samples and the training subset is filtered by removing the training samples according to the prediction scores. Concretely, the training subset is updated using a probabilistic sampling algorithm to remove the instances whose prediction scores are near 0.5 with a high probability. In other words, we intend to keep the samples with high distinguishability in the training subset. Let $s=\{s_1,s_2\}$ be the prediction score of a sample in the training set where s_1 denotes the probability that the sample belongs to the "positive" class and s_2 denotes the probability that the sample belongs to the "negative" class. Since the prediction scores are between the range of 0 to 1 and the sum of them is 1, we can take the scores as the probabilities of an information source. Here we propose an algorithm based on the entropy theory [19] by removing a training sample with a probability of p:

$$p = H(s) = -(s_1 \log_2 s_1 + s_2 \log_2 s_2) \tag{8}$$

where $H(s)$ is the entropy of the prediction scores.

When the difference between the sentiment scores of a training instance becomes too small or the two prediction scores get close to 0.5, $H(s)$ reaches its maximum in the most uncertain situation. Then we remove this instance from the training subset with a larger probability. Oppositely, we will keep this training sample with a large probability when the difference is large enough. Finally, we use the updated dataset to fine-tune the model and choose the final model to predict visual sentiment.

4. Experiments

We first trained the NIN we proposed with the Columbia University's dataset [4] which consists of a set of Flickr images corresponding to 1553 ANPs. Since each ANP has been assigned a sentiment value between -2 (negative) and +2 (positive), we used the binary sentiment values as the labels of images in each ANP subset directly. Additionally, we evaluated the performance of our model on another image dataset [14] with a total of 1269 images from image tweets. In order to make a comparison with the previous work in [14], we also employed transfer learning to further fine-tune our model with the manually

labeled images in this dataset. The proposed architecture of NIN was implemented on a publicly available platform called Caffe [20].

4.1. Training on Flickr Dataset

The model shown in Figure 3, was trained with about 90% of the Flickr images (about 400,000 images) provided in [4] and tested with the remaining images (about 44,000 images) of the dataset. The minimization process of the regression objective was carried out by stochastic gradient descent with a momentum of 0.9 and weight decay of 0.0005. The learning rate was initialized at 0.01 and the NIN was trained with 300,000 iterations (about 190 epochs) of mini-batches, each of which contains 256 images. The training dataset was further filtered with the sampling probability described in Equation 8. About 30,000 images were excluded from the training dataset after sampling. Finally, the model was trained with another 100,000 iterations (about 70 epochs) using the filtered training dataset in the progressive fine-tuning stage.

Table 1. Performance on Flickr Dataset Using NIN and PNIN

Algorithm	Precision	Recall	F1	Accuracy
NIN	0.747	0.764	0.755	0.753
PNIN	0.795	0.864	0.826	0.812

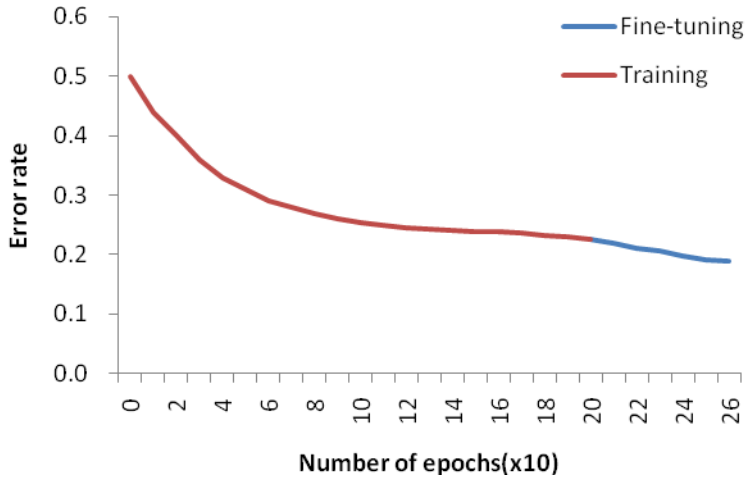


Figure 5. Test Error of NIN with Fine-Tuning in the Total 260 Epochs of Training (The Curve Has Been Smoothed)

Figure 5, shows the test error of NIN with progressive fine-tuning in the training process. Even if we can't make a conclusion that the neural network has got a better local optimum with the fine-tuning stage of PNIN, we can at least learn from the curves in Figure 5, that the progressive fine-tuning with less noisy training examples can provide different knowledge for the neural network. Additionally, Table 1, gives the performance for both NIN and PNIN in terms of Precision, Recall, F1 and Accuracy. Experimental results indicate that PNIN outperforms NIN with as much as 6~13% gain on all performance indicators with a 6.4% increment on precision, 13.1% on recall, 9.4% on F1, and 7.8% on accuracy. All the evaluation results suggest that the model achieves significant performance improvement after fine-tuning with cleaner training samples.

4.2. Twitter Test Dataset

We also evaluated the performance of NIN for sentiment prediction on a benchmark containing 1269 photo tweets covering a wide range of topics [14]. The labels of these test images were manually obtained by Amazon Mechanic Turk (AMT) annotation which is a kind of crowd intelligence techniques. Five AMT workers were recruited to give their sentiment labels for each candidate image. There are 882 "five agree" images to which all the five AMTs gave the same sentiment labels. Similarly, there are 1116 "at least four agree" images and 1269 "at least three agree" images.

As shown in Table 2, and Figure 6, we compare the performance of the proposed framework and traditional CNNs on the twitter dataset in terms of Precision, Recall, F1 and Accuracy. It is obvious that NIN models perform better than CNN models on all the three labeling sets mentioned above. For example, PNIN outperforms PCNN by about 4% on the "five agree" image set, about 4.5% on the "at least four agree" image set and about 5% on the "at least three agree" image set. At the same time, it can be observed that PNIN outperforms NIN just as PCNN outperforms CNN in most cases. Thus the results further indicate that the progressive fine-tuning stage can improve the performance of the neural network effectively.

Table 2. Performance of Different Models on the Twitter Image Dataset(Prec Stands for Precision, Rec Stands for Recall and Acc Stands for Accuracy)

Models	Five agree				At least four agree				At least three agree			
	Prec	Rec	F1	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1	Acc
CNN[14]	0.749	0.869	0.805	0.722	0.707	0.839	0.768	0.686	0.691	0.814	0.747	0.667
PCNN[14]	0.770	0.878	0.821	0.747	0.733	0.845	0.785	0.714	0.714	0.806	0.757	0.687
NIN	0.778	0.899	0.836	0.750	0.738	0.876	0.803	0.716	0.726	0.855	0.784	0.702
PNIN	0.801	0.903	0.854	0.776	0.765	0.883	0.819	0.745	0.751	0.847	0.795	0.721

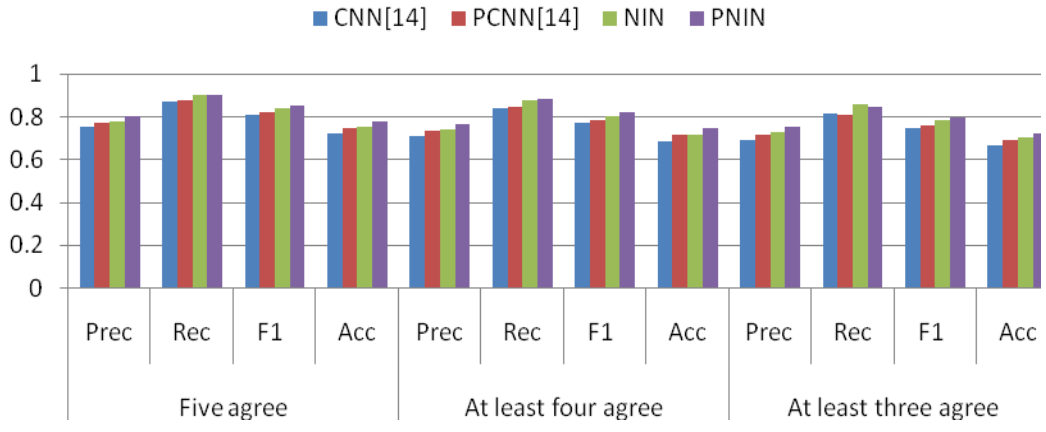


Figure 6. Comparison of the Performance of Different Models on the Twitter Image Dataset(Prec Stands for Precision, Rec Stands for Recall and Acc Stands for Accuracy)

4.3. Transfer Learning

Experiments in Section 4.2, indicate that the generic features learned from Flickr images can be utilized to predict visual sentiment of Twitter images from another domain. For the reason that Twitter images are mostly related to personal experiences and hot topics, they have greater diversity in content and quality. This raises the possibility for adopting transfer learning to further improve the model's performance on Twitter images.

Here, we employed a method similar to [14] to achieve transfer learning as follows. We first equally divided the images from Twitter into five partitions before fine-tuning. Each time, we used four of the five partitions to further fine-tune the model which had been trained with the Flickr images dataset and then took the remaining partition as the test dataset to evaluate the performance of the newly trained model. This process was repeated five times to make sure each partition had been used as the test dataset once. We finally took the averaged results to evaluate the performance of all the models. This strategy can be called 5-fold cross-validation which was also adopted in [3] and [14].

Table 3. 5-Fold Cross-Validation Performance of NIN Models on the Twitter Image Dataset (Prec Stands for Precision, Rec Stands for Recall and Acc Stands for Accuracy)

Models	Five agree				At least four agree				At least three agree			
	Prec	Rec	F1	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1	Acc
NIN	0.824	0.929	0.876	0.812	0.806	0.892	0.846	0.787	0.770	0.872	0.816	0.752
PNIN	0.827	0.904	0.871	0.803	0.819	0.878	0.845	0.791	0.793	0.845	0.814	0.758

Table 3, gives the averaged performance results of the NIN models we proposed. Compared to the results reported in Table 2, the performance of both NIN models has been significantly improved by transfer learning on all the three sets of the Twitter testing data. This indicates that transfer learned knowledge from the Twitter images causes both NIN models to reach a better local minimum and influences the performance of both networks. This therefore suggests that it is feasible to transfer the knowledge from one domain to another with simple fine-tuning techniques when we adopt convolutional neural networks to predict visual sentiment. This will greatly improve efficiency for the reason that we don't always have to start training neural networks from scratch.

5. Conclusions

In this paper, we introduce a recently developed convolutional neural network (NIN) into visual sentiment analysis. Additionally, we further optimize the neural network by filtering the noise in the pseudo labeled training images in a progressive way. We propose an algorithm based on entropy theory to remove the training samples with ambiguous sentiment prediction. Extensive experiments reveal that the new structure which adopts multilayer perceptrons to convolve the input performs better than traditional CNNs for the challenging problem of visual sentiment analysis. Furthermore, experiments using transfer learning indicate the feasibility of acquiring knowledge from other domains. In one word, our work shows the possibility of introducing deep learning algorithms in computer vision to visual sentiment analysis. We believe that the active researches in deep learning will provide promising opportunities for this novel research area.

Acknowledgments

This work was supported by the Science and Technology Innovation Engineering Program for Shaanxi Provincial Key Laboratories under Grant 2013SZS15-K02.

References

[1] B. Pang and L. Lee, "Opinion Mining and Sentiment Analysis", Information Retrieval, vol. 2, no. 1-2, **(2008)**, pp. 1-135.
[2] T. Chen, D. Borth, T. Darrell and S. F. Chang, "DeepSentiBank: Visual Sentiment Concept Classification with Deep Convolutional Neural Networks", arXiv preprint arXiv: 1410.8586, **(2014)**.
[3] D. Borth, R. Ji, T. Chen, T. Breuel and S. F. Chang, "Large-scale visual sentiment ontology and detectors using adjective noun pairs", Proceedings of the 21st ACM international conference on Multimedia; Barcelona:Spain, **(2013)** October 21-25.
[4] D. Borth, T. Chen, R. Ji and S. F. Chang, "SentiBank: large-scale ontology and classifiers for detecting sentiment and emotions in visual content", Proceedings of the 21st ACM international conference on Multimedia; Barcelona:Spain, **(2013)** October 21-25.
[5] M. Thelwall, K. Buckley, G. Paltoglou, D. Cai and A. Kappas, "Sentiment strength detection in short informal text", Journal of the American Society for Information Science and Technology, vol. 61, no. 12, **(2010)**, pp. 2544-2558.
[6] J. Machajdik and A. Hanbury, "Affective Image Classifcation using Features inspired by Psychology and Art Theory", Proceedings of the international conference on Multimedia; Florence: Italy, **(2010)** October 25-29.
[7] J. Jia, S. Wu, X. Wang, P. Hu, L. Cai and J. Tang, "Can we Understand van Gogh's Mood?Learning to infer Affects from Images in Social Networks", Proceedings of the 20th ACM international conference on Multimedia,**(2012)** October 29-November 02; Nara:Japan.
[8] S. Siersdorfer, E. Minack, F. Deng and J. Hare, "Analyzing and predicting sentiment of images on the social web", Proceedings of the international conference on Multimedia; Florence: Italy, **(2010)** October 25-29.
[9] J. Yuan, S. Mcdonough, Q. You and J. Luo, "Sentribute:image sentiment analysis from a mid-level perspective", Proceedings of the Second International Workshop on Issues of Sentiment Discovery and Opinion Mining; Chicago:USA, **(2013)** August 11-14.
[10] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition", Neural Computation , vol. 1, no. 4, **(1989)**, pp. 541–551.
[11] G. E. Hinton, S. Osindero and Y. W. Teh, "A fast learning algorithm for deep belief nets", Neural Computation, vol. 18, no. 7, **(2006)**, pp. 1527–1554.
[12] D. C. Ciresan, U. Meier, J. Masci, L. M. Gambardella and J. Schmidhuber, "Flexible, high performance convolutional neural networks for image classification", Proceedings of International Joint Conference on Artificial Intelligence; Barcelona: Spain, **(2011)** July 16-22.
[13] A. Krizhevsky, I. Sutskever and G. E. Hinton, "Imagenet classification with deep convolutional neural networks", Proceedings of Advances in neural information processing systems; Lake Tahoe:USA, **(2012)** December 3-8.
[14] Q. You, J. Luo, H. Jin and J. Yang, "Robust Image Sentiment Analysis using Progressively Trained and Domain Transferred Deep Networks", Proceedings of The Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI) ; Austin:USA, **(2015)** January 25–30.
[15] M. Lin, Q. Chen and S. Yan, "Network in Network", arXiv preprint arXiv: 1312.4400v3, **(2014)**.
[16] Y. LeCun, L. Bottou, Y. Bengio and P. Haffner, "Gradient-based learning applied to document recognition", Proceedings of the IEEE, vol. 86, no. 11, **(1998)**, pp. 2278-2324.
[17] Y. Bengio, A. Courville and P. Vincent, "Representation learning: A review and new perspectives", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 35, no. 8, **(2013)**, pp. 1798–1828.
[18] R. Plutchik, "Emotions: A general psychoevolutionary theory", Approaches to Emotion, **(1984)**, pp. 197–219.
[19] C. E. Shannon, "A mathematical theory of communication", ACM SIGMOBILE Mobile Computing and Communications Review, vol. 5, no. 1, **(2001)**, pp. 3-55.
[20] Y. Jia, "Caffe: An open source convolutional architecture for fast feature embedding", http://caffe.berkeleyvision.org/, **(2013)**.

Authors

Zuhe Li received the B.S. degree in electronic information science and technology from Zhengzhou University of Light Industry, Zhengzhou, China, in 2004, and the M.S. degree in communication and information system from Huazhong University of Science and Technology, Wuhan, China, in 2008.He is currently pursuing the Ph.D. degree at the Northwestern Polytechnical University, Xi'an, China. His major research interests include Computer Vision and Machine Lerning.

Yangyu Fan received the B.S. degree and M.S. degree from Shaanxi University of Science & Technology, Xi'an, China, in 1982 and 1992 respectively, and the Ph.D. degree from Northwestern Polytechnical University, Xi'an, China, in1999. He is the director of the Laboratory of Multimedia and Virtual Reality of School of Electronics and Information, Northwestern Polytechnical University. His research interests include Computer Vision, Virtual Reality and Signal Processing.

Fengqin Wang received the B.S. degree from Zhengzhou University, Zhengzhou, China, in 2002, and the M.S. degree and Ph.D. degree from Northwestern Polytechnical University, Xi'an, China, in 2005 and 2010 respectively. Her research interests include Video Coding, Virtual Reality and Signal Processing.

