On Estimating Air Pollution from Photos Using Convolutional Neural Network

Chao Zhang
Beijing Normal University
IBM Research - China
zhangch@mail.bnu.edu.cn

Junchi Yan
East China Normal University
IBM Research - China
yanjc@cn.ibm.com

Changsheng Li*
IBM Research - China
lcsheng@cn.ibm.com

Xiaoguang Rui
IBM Research - China
ruixg@cn.ibm.com

Liang Liu
IBM Research - China
liuliang@cn.ibm.com

Rongfang Bie
Beijing Normal University
rfbie@bnu.edu.cn

ABSTRACT

Air pollution has raised people's intensive concerns especially in developing countries such as China and India. Different from using expensive or unreliable methods like sensor-based or social network based one, photo based air pollution estimation is a promising direction, while little work has been done up to now. Focusing on this immediate problem, this paper devises an effective convolutional neural network to estimate air's quality based on photos. Our method is comprised of two ingredients: first a negative log-log ordinal classifier is devised in the last layer of the network, which can improve the ordinal discriminative ability of the model. Second, as a variant of the Rectified Linear Units (ReLU), a modified activation function is developed for photo based air pollution estimation. This function has been shown it can alleviate the vanishing gradient issue effectively. We collect a set of outdoor photos and associate the pollution levels from official agency as the ground truth. Empirical experiments are conducted on this real-world dataset which shows the capability of our method.

1. INTRODUCTION

Air pollution is becoming more and more serious in many developing countries, which influences the human health and life significantly [22]. Thus, how to reduce the air pollution becomes an urgent problem in these countries. In order to achieve this goal, air quality estimation is an effective way to monitor the air pollution and can provide recommendations for decision makers. In the past, many methods have been proposed for predicting air quality. Chung [5] utilizes the satellites to detect the air pollution and its transport. Hodgeson et al. [10] use the spectroscopic technique to solve the complexity of the monitoring problems. Other works include filter-based gravimetric method [7], optical information of the atmospheric aerosol [23], social network based method [4], and mobile information based method [19]. However traditional air quality monitoring needs professional sensors or equipments deployed at representative sites, which is expensive in the cost. More importantly, their coverage is geographically limited, and thus the air pollution levels in those places far from air quality monitor sites can not be measured accurately. With the rapid development of smartphone, directly estimating air pollution from photos starts to gain the potential of being a convenient and less expensive approach because it can cover more areas in a crowd-sourcing manner. However, little work has been done for photo based air pollution estimation.

Feature learning based on convolutional neural networks (CNN) has been intensively studied in recent years, and many promising progresses have been made for solving typical computer vision problems [12]. In CNN based methods, designing a task-specific network structure is rather challenging. In this paper, we explore the way for how to design a CNN model for estimating air pollution by photos from mobile-phone. In particular, this paper focuses on two aspects in the CNN model design. First, one popular classifier in CNNs is Softmax [9] which is the generalization of the logistic regression. In addition, Softmax regression is a supervised learning algorithm, which can utilize the features derived from deep learning networks as input for prediction. Such combination has achieved prominent success in many tasks such as MNIST digit, CIFAR10 and CIFAR100 classification. However multinomial regression model like Softmax does not consider the ranking information at all during learning while ranking information is very important for ranking or regression problems [14, 15]. Therefore, designing a network structure to preserve the ranking information is necessary for air quality prediction.

Second, the activation function is another critical part for the success of CNN applications. The sigmoid suffers from the vanishing gradient problem [3]. The mean of ReLU [20] activation is larger than zero which may lead to bias for the next layer. PReLU [8] and LReLU [18] consider the nonzero slope for negative part but they do not consider change the slope of the positive part to improve model's flexibility.

In this paper, we propose a CNN based Photo Air Pollution Level Estimation algorithm, called PAPLE, for air pollution estimation. Specifically, a new function with two parameters $\theta = \{\alpha, \beta\}$ is first introduced as the activation operator followed by each convolutional layer. The positive parts α is

*Corresponding author.

MM '16, October 15 - 19, 2016, Amsterdam, Netherlands

ISBN 978-1-4503-3603-1/16/10...$15.00

DOI: http://dx.doi.org/10.1145/2964284.2967230

the slope and the negative part $\beta < 0$ is the constant term. The new activation layer enables our CNN model to be more flexible, less time-consuming and overcome the bias problem for the next layer. In the proposed convolutional neural network model, there are nine convolution layers, two pooling layers and two dropout layers. To capture the ranking information of the categories, we then involve in the ordinal classifier [6] in the last layer of network. Because most day's pollution levels are low and thus the cumulative probability for lower parts is high, we choose negative log-log as our link function [21].

In a nutshell, the main bullets of our method are:

- This paper is one of the first work for estimating the air pollution via image content. We collect a labeled dataset involving both photos taken in different air conditions and the pollution measurements from official agency. Empirical results show the feasibility of image based pollution estimation.

- A novel convolutional neural network is tailored to the air pollution estimation task with two ingredients. First, a modified activation function is introduced to effectively mitigate the problem of vanishing gradient and the bias for the next layer. Second, instead of the widely used Softmax classifier, a negative log-log ordinal classifier is adopted to better fit the pollution estimation problem.

2. POLLUTION ESTIMATION BY CNN

2.1 Convolutional Features Extractor

We first present the CNN features used in this work. The operator of the convolution layer can be summarized as

$$L_i = pool_P(\sigma(L_{i-1} * W_i + b_i)) \tag{1}$$

where L_{i-1} is the input feature layer, W_i are the weights and b_i are the biases. $\theta_i = \{W_i, b_i\}$ is the learned parameter set. The function *pool* is a sub-sampling operation. P is the size of the pooling region. $*$ represents the convolution operator, and σ is the activation function.

ReLU [20] has been proposed to replace saturated counterpart (e.g. sigmoid and tanh) activation function and achieves excellent performance. However, ReLU's mean activation is larger than zero which may lead to bias for the next layer. And ReLU's slope is 1 which is not flexible. Thus, we define a new activation as followed:

$$f(x) = \begin{cases} \beta, & if\ x < 0 \\ \alpha x, & otherwise \end{cases} \tag{2}$$

where $\theta = \{\alpha, \beta\}$ are the parameters. α is the slope and $\beta < 0$ is the constant term. The new activation only has two parameters and it is fast to train the model. Our proposed activation is similar to other ReLU family activation, like PReLU [8], LReLU [18]. But our function has one negative constant value. This simplifies our model and the slope parameter for positive part renders our function more flexible. Specifically when $\alpha = 1$ and $\beta = 0$, our activation function is reduced to standard ReLU. Fig.1 shows the shape of ReLU and our proposed activation function. The first derivative is derived as

$$f'(x) = \begin{cases} 0, & if\ x < 0 \\ \alpha, & otherwise \end{cases} \tag{3}$$

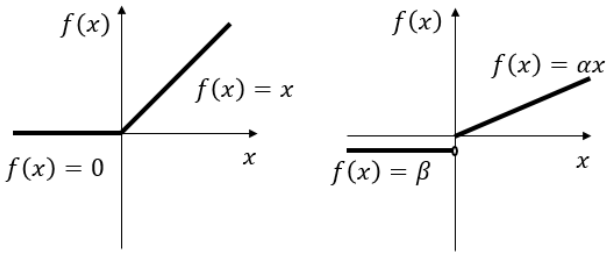


Figure 1: ReLU vs. the proposed activation function. The parameter $\theta = \{\alpha, \beta\}$, where α is the slope and β is the negative part.

The new activation function can be trained by backpropagation [13], the update rule for $\theta = \{\alpha, \beta\}$ is obtained from chain rule:

$$\frac{\partial O}{\partial \theta} = \sum_{x_i} \frac{\partial O}{\partial f(x_i)} \frac{\partial f(x_i)}{\partial \theta} \tag{4}$$

where O denotes the objective function, $\frac{\partial O}{\partial f(x_i)}$ is the gradient propagated from the higher layers. For the channel-shared variant, the gradient of θ is $\frac{\partial O}{\partial \theta} = \sum_i \sum_{x_i} \frac{\partial O}{\partial f(x_i)} \frac{\partial f(x_i)}{\partial \theta}$.

The gradients for the parameters α and β are

$$\frac{\partial f(x_i)}{\partial \alpha} = \begin{cases} 0, & if\ x < 0 \\ x_i, & otherwise \end{cases} \tag{5}$$

$$\frac{\partial f(x_i)}{\partial \beta} = \begin{cases} 1, & if\ x < 0 \\ 0, & otherwise \end{cases} \tag{6}$$

The momentum algorithm is used to update the $\theta = \{\alpha, \beta\}$

$$\Delta\theta := \mu\Delta\theta + \epsilon\frac{\partial O}{\partial \theta} \tag{7}$$

where μ is the momentum and ϵ is the learning rate.

To obtain features from CNN, we introduce a CNN based network structure: Estimation of Photo Air Pollution Level Network (EPAPLN). Its structure is summarized in Table 1. Our convolutional neural network includes 9 convolution layers, 2 pooling layers and 2 dropout layers. Our proposed activation function is used in each convolutional layer. In each convolution layer the first part shows the filter shape and the second part shows number of filters. The size of filters of pooling layers is set to 3×3. We utilize dropout [24] as the regularization method to prevent neural networks from overfitting. We set the probability of retaining a hidden unit to 0.5.

Stochastic gradient descent is applied to learn EPAPLN. The last two layers are used to represent the features because the high-level layers are more related to semantics [17].

2.2 Negative Log-log Ordinal Classifier

Softmax ignores data's order and treats all the categorical variables as nominal ones, which introduces bias for ordinal regression tasks. Hence we adopt the ordinal classifier to estimate the pollution level. Because pollution levels for most days are low and the cumulative probability for lower parts is high, we choose negative log-log as our link function. The negative log-log ordinal regression model is:

$$-\log\left(-\log\left(P(Y \leqslant j)\right)\right) = \alpha_j + \sum \omega_i x_i \tag{8}$$

The log-likelihood function can be derived as below:

$$\ell = \sum_{j=1}^{J-1} \sum_{k=1}^{j} nk \log(\frac{\sum_l^j \pi_l}{\pi_{j+1}}) - \sum_{k=1}^{j+1} nk \log(\frac{\sum_l^{j+1} \pi_l}{\pi_{j+1}}) \tag{9}$$

Table 1: Network structure: each layer is a convolutional layer if not otherwise specified. Activation function is followed by each convolutional layer.

Input Size	EPAPLN (our model)
64×64	3×3, 32
64×64	3×3, 32
64×64	3×3, max pooling, /2
32×32	dropout, 0.5
16×32	3×3, 64
32×32	3×3, 64
32×32	3×3, 64
32×32	3×3, avg pooling, /2
16×8	dropout, 0.5
16×16	3×3, 96
16×16	3×3, 96
16×16	3×3, 96
16×16	3×3, 96

where n is the sum of all frequency weights. We denote the parameter vector as θ, which has $J-1+p$ parameters. $J=6$ is the level count and p is the feature number.

Newton-Raphson method can be used to estimate the parameters and let $theta_s$ be the parameter vector at step s and let A_s be a $(J-1+p)\times(J-1+p)$ matrix.

$$[A_s]_d e = -\frac{\partial^2 \ell}{\partial\theta_d \partial\theta_e} \tag{10}$$

For step $s+1$, the parameter vector θ_{s+1} can be obtained by:

$$A_s\theta_{s+1} = A_s\theta_s + \gamma\frac{\partial\ell}{\partial\theta_s} \tag{11}$$

where γ is the stepping scalar.

The first derivative of the log-likelihood regarding θ is:

$$\begin{aligned}\frac{\partial\ell}{\partial\theta_d} = \sum_{j=1}^{J-1}\left(\sum_{k=1}^{j} nk - \sum_{k=1}^{j+1} nk\frac{\sum_{l=1}^{j}\pi_l}{\sum_{l=1}^{j+1}\pi_l}\right)\frac{\sum_{l=1}^{j+1}\pi_l}{\sum_{l=1}^{j}\pi_l\pi_{j+1}} \\ \left(\frac{\partial(\alpha_{j+1}+\sum\omega_i x_i)}{\partial\theta_d}\frac{\sum_{l=1}^{j}\pi_l}{\sum_{l=1}^{j+1}\pi_l}\sum_{l=1}^{j+1}\pi_l\log(\sum_{l=1}^{j+1}\pi_l)\right. \\ \left. -\frac{\partial(\alpha_j+\sum\omega_i x_i)}{\partial\theta_d}\sum_{l=1}^{j}\pi_l\ \log(\sum_{l=1}^{j}\pi_l)\right)\end{aligned} \tag{12}$$

where we set $\frac{\partial(\alpha_J+\sum\omega_i x_i)}{\partial\theta} = 0$, $\sum_{l=1}^{J}\pi_l\log(\sum_{l=1}^{J}\pi_l) = 0$ and $j=1,\cdots,J-1$, $d=1,\cdots,J-1+p$.

The second derivative of the log-likelihood w.r.t. θ is:

$$\frac{\partial^2\ell}{\partial\theta_d\partial\theta_e} = \sum_{j=1}^{J-1}(F_1 - F_2 + F_3F_4) \tag{13}$$

where F_1, F_2, F_3, F_4 can be computed as follows:

$$\begin{aligned}F_1 = Q_{je}\left(\sum_{k=1}^{j+1} nk\frac{\sum_{l=1}^{j}\pi_l}{\sum_{l=1}^{j+1}\pi_l} - \sum_{k=1}^{j} nk\right) \\ \left(\frac{1}{(\sum_{l=1}^{j}\pi_l)^2}\frac{\sum_{l=1}^{j+1}\pi_l}{\sum_{l=1}^{j}\pi_l\pi_{j+1}}Q_{jd}\right. \\ \left. +\frac{1}{\pi_{j+1}^2}\left(\frac{\sum_{l=1}^{j+2}\pi_l}{\sum_{l=1}^{j+1}\pi_l\pi_{j+2}}Q_{(j+1)d} - \frac{\sum_{l=1}^{j+1}\pi_l}{\sum_{l=1}^{j}\pi_l\pi_{j+1}}Q_{jd}\right)\right)\end{aligned} \tag{14}$$

$$F_2 = \frac{\sum_{k=1}^{j+1} nk}{\sum_{l=1}^{j+1}\pi_l}\frac{\sum_{l=1}^{j+1}\pi_l}{\sum_{l=1}^{j}\pi_l\pi_{j+1}}Q_{jd}Q_{je} \tag{15}$$

$$F_3 = -\left(\sum_{k=1}^{j+1} nk\frac{\sum_{l=1}^{j}\pi_l}{\sum_{l=1}^{j+1}\pi_l} - \sum_{k=1}^{j} nk\right)\frac{\sum_{l=1}^{j+1}\pi_l}{\sum_{l=1}^{j}\pi_l\pi_{j+1}} \tag{16}$$

$$\begin{aligned}F4 = \frac{\partial(\alpha_j+\sum\omega_i x_i)}{\partial\theta_e}\left(1+\log\sum_{l=1}^{j}\pi_l\right) \\ \sum_{l=1}^{j}\pi_l\log(\sum_{l=1}^{j}\pi_l)\frac{\partial(\alpha_j+\sum\omega_i x_i)}{\partial\theta_d} \\ +\frac{\partial(\alpha_{j+1}+\sum\omega_i x_i)}{\partial\theta_e}\frac{Q_{jd}}{\sum_{l=1}^{j+1}\pi_l}\sum_{l=1}^{j+1}\pi_l\log(\sum_{l=1}^{j+1}\pi_l) \\ +\frac{\partial(\alpha_{j+1}+\sum\omega_i x_i)}{\partial\theta_d}\frac{\sum_{l=1}^{j}\pi_l}{\sum_{l=1}^{j+1}\pi_l}\left(1+\log\sum_{l=1}^{j+1}\pi_l\right) \\ \sum_{l=1}^{j+1}\pi_l\log(\sum_{l=1}^{j+1}\pi_l)\frac{\partial(\alpha_{j+1}+\sum\omega_i x_i)}{\partial\theta_d}\end{aligned} \tag{17}$$

where $d, e = 1,\cdots,(J-1)+p$ and Q_{jd} and Q_{je} are below:

$$\begin{aligned}Q_{jd} = \left(\frac{\partial(\alpha_{j+1}+\sum\omega_i x_i)}{\partial\theta_d}\frac{\sum_{l=1}^{j}\pi_l}{\sum_{l=1}^{j+1}\pi_l}\sum_{l=1}^{j+1}\pi_l\log(\sum_{l=1}^{j+1}\pi_l)\right. \\ \left. -\frac{\partial(\alpha_j+\sum\omega_i x_i)}{\partial\theta_d}\sum_{l=1}^{j}\pi_l\log(\sum_{l=1}^{j}\pi_l)\right)\end{aligned} \tag{18}$$

$$\begin{aligned}Q_{je} = \left(\frac{\partial(\alpha_{j+1}+\sum\omega_i x_i)}{\partial\theta_e}\frac{\sum_{l=1}^{j}\pi_l}{\sum_{l=1}^{j+1}\pi_l}\sum_{l=1}^{j+1}\pi_l\log(\sum_{l=1}^{j+1}\pi_l)\right. \\ \left. -\frac{\partial(\alpha_j+\sum\omega_i x_i)}{\partial\theta_e}\sum_{l=1}^{j}\pi_l\log(\sum_{l=1}^{j}\pi_l)\right)\end{aligned} \tag{19}$$

3. EXPERIMENTS

We evaluate our Photo Air Pollution Level Estimation (PAPLE) algorithm on the pollution images dataset which includes photos shots taken in Beijing by ourself. The dataset involves two major air pollutants in developing countries including Particulate Matter 2.5 (PM2.5) and Particulate Matter 10 (PM10) as the main targets for estimation, which have both 6 levels [1] as listed in Table 2. The associated ground truth, i.e. the air quality observation data, is collected from the website of Beijing environmental protection bureau [2].

Table 2: Air quality index for PM2.5 and PM10.

Level	1	2	3	4	5	6
PM2.5 Range	<35	35-75	75-115	115-150	150-250	>250
PM10 Range	<50	50-150	150-250	250-350	350-420	>420

We compare CNN-Softmax, PReLU-CNN-Softmax, LReLU-CNN-Softmax, ReLU-CNN-Softmax, PCA-Negative Log-log Ordinal Classifier (PCALL) and use their default parameters to train the model. Table 3 shows our proposed PAPLE performs competitively against the other methods on the

Table 4: Coincidence matrix for PM2.5 by different methods on the collected real-world dataset.

	1	2	3	4	5	6
1	326	13	3	1	3	2
2	63	48	11	1	0	4
3	33	23	23	8	4	1
4	11	10	4	5	8	3
5	15	4	2	10	43	9
6	3	2	2	4	12	44

(PAPLE)

	1	2	3	4	5	6
1	305	25	14	0	4	0
2	57	47	19	1	2	1
3	25	25	30	6	6	0
4	7	8	7	5	9	5
5	8	3	22	5	30	15
6	8	0	4	0	18	37

(CNN-Softmax)

	1	2	3	4	5	6
1	263	50	26	5	4	0
2	38	45	27	13	3	1
3	17	17	30	19	7	2
4	5	6	11	5	6	8
5	4	6	12	12	32	17
6	0	3	4	3	20	37

(PReLU-CNN-Softmax)

	1	2	3	4	5	6
1	296	44	0	0	8	0
2	53	68	0	0	5	1
3	28	56	0	0	7	1
4	4	22	0	0	13	2
5	6	24	0	0	46	7
6	5	5	0	0	28	29

(LReLU-CNN-Softmax)

	1	2	3	4	5	6
1	293	35	3	6	6	5
2	61	45	9	1	2	9
3	30	25	14	10	6	7
4	11	8	4	6	9	3
5	15	5	0	11	38	14
6	3	1	2	1	15	45

(ReLU-CNN-Softmax)

	1	2	3	4	5	6
1	293	46	4	0	5	0
2	62	59	2	0	3	1
3	37	43	6	0	5	1
4	7	21	0	0	11	2
5	6	30	2	0	37	8
6	2	3	5	0	27	30

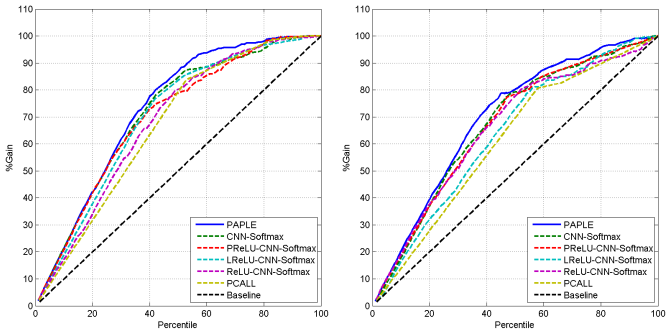
(PCALL)

Table 3: Average error with different methods.

Methods	PM2.5	PM10
PAPLE	0.606	0.411
CNN-Softmax	0.644	0.47
PReLU-CNN-Softmax	0.691	0.492
LReLU-CNN-Softmax	0.694	0.511
ReLU-CNN-Softmax	0.699	0.526
PCALL	0.724	0.569

average error which is defined by: $(\sum_{i=1}^{n} |predicted\ level - actual\ level|)/n$, where n is the number of testing samples. Table 4 shows the coincidence matrices which show the pattern of matches between each predicted level and its actual level with different methods for the PM2.5 as target. The rows refer to the actual PM2.5 levels, the columns show the predicted PM2.5 levels and the content of table shows the matching number of record. Larger diagonal numbers indicate better performance. Because of the space limitation, we do not list the PM10 results. In fact they are similar to the PM2.5 results.

Cumulative gain [11] is also adopted to measure the performance. It is calculated as the ratio between the results obtained with and without the model and it denotes the percentage of the overall number of cases in a given category "gained" by targeting a percentage of the total number of cases: $Gain = \frac{Expected\ Response\ Using\ Predictive\ Model}{Expected\ Response\ From\ Random\ Mailing}$. In Figure 2, y-axis shows the percentage of positive responses, and the result demonstrates PAPLE still outperforms all the other methods in terms of this metric.



PM2.5 PM10

Figure 2: Cumulative gains for PM2.5, PM10.

4. CONCLUSION

In this paper, we have proposed an effective CNN based model for air pollution estimation from raw images. Our model involves a negative log-log ordinal classifier to fit the ordinal output well. We also empirically devise a new activation function for photo air pollution level estimation. The proposed approach is validated with qualitative and quantitative evaluations against several state-of-the-art methods. One important future work is to use crowd-sourced photos to trace contamination path. We will collect more photos from different places to study the influence of the diversity between the training and the testing dataset. We will also consider the image quality and noises in our model [16, 25].

5. ACKNOWLEDGMENT

This research is sponsored by National Natural Science Foundation of China (No.61171014, 61371185, 61401029, 11401028, 61571049) and the Fundamental Research Funds for the Central Universities (No.2014KJJCB32, 2013NT57, 2012LY-B46) and by SRF for ROCS, SEM and China Postdoctoral Science Foundation Funded Project (No.2016M590337).

6. REFERENCES

[1] http://210.72.1.216:8080/gzaqi/Document/gjzlbz.pdf. Accessed: 2016-07-19.

[2] http://zx.bjmemc.com.cn/. Accessed: 2016-07-19.

[3] Y. Bengio, P. Simard, and P. Frasconi. Learning long-term dependencies with gradient descent is difficult. *Neural Networks, IEEE Transactions on*, 5(2):157–166, 1994.

[4] J. Chen, H. Chen, G. Zheng, J. Z. Pan, H. Wu, and N. Zhang. Big smog meets web science: smog disaster analysis based on social media and device data on the web. In *Proceedings of the companion publication of the 23rd international conference on World wide web companion*, pages 505–510. International World Wide Web Conferences Steering Committee, 2014.

[5] Y. Chung. Air pollution detection by satellites: The transport and deposition of air pollutants over oceans. *Atmospheric Environment (1967)*, 20(4):617–630, 1986.

[6] S. Greenland. Alternative models for ordinal logistic regression. *Statistics in medicine*, 13(16):1665–1677, 1994.

[7] H. Hauck, A. Berner, B. Gomiscek, S. Stopper, H. Puxbaum, M. Kundi, and O. Preining. On the equivalence of gravimetric pm data with teom and beta-attenuation measurements. *Journal of Aerosol Science*, 35(9):1135–1149, 2004.

[8] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1026–1034, 2015.

[9] G. E. Hinton and R. R. Salakhutdinov. Replicated softmax: an undirected topic model. In *Advances in neural information processing systems*, pages 1607–1614, 2009.

[10] J. Hodgeson, W. McClenny, and P. Hanst. Air pollution monitoring by advanced spectroscopic techniques a variety of spectroscopic methods are being used to detect air pollutants in the gas phase. *Science*, 182(4109):248–258, 1973.

[11] K. J. and J. K. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems*, 20:2002, 2002.

[12] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[13] 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*, 1(4):541–551, 1989.

[14] C. Li, Q. Liu, J. Liu, and H. Lu. Learning ordinal discriminative features for age estimation. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2570–2577, 2012.

[15] C. Li, Q. Liu, J. Liu, and H. Lu. Ordinal distance metric learning for image ranking. *IEEE transactions on neural networks and learning systems*, 26(7):1551–1559, 2015.

[16] Y. Li, Y. Zhou, J. Yan, J. Yang, and X. He. Tensor error correction for corrupted values in visual data. In *2010 IEEE International Conference on Image Processing*, pages 2321–2324. IEEE, 2010.

[17] C. Ma, J.-B. Huang, X. Yang, and M.-H. Yang. Hierarchical convolutional features for visual tracking. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3074–3082, 2015.

[18] A. L. Maas, A. Y. Hannun, and A. Y. Ng. Rectifier nonlinearities improve neural network acoustic models. In *Proc. ICML*, volume 30, page 1, 2013.

[19] S. Mei, H. Li, J. Fan, X. Zhu, and C. R. Dyer. Inferring air pollution by sniffing social media. In *Advances in Social Networks Analysis and Mining (ASONAM), 2014 IEEE/ACM International Conference on*, pages 534–539. IEEE, 2014.

[20] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 807–814, 2010.

[21] J. A. Nelder and R. J. Baker. Generalized linear models. *Encyclopedia of Statistical Sciences*, 1972.

[22] C. A. Pope III and D. W. Dockery. Health effects of fine particulate air pollution: lines that connect. *Journal of the air & waste management association*, 56(6):709–742, 2006.

[23] J. D. Smith and D. B. Atkinson. A portable pulsed cavity ring-down transmissometer for measurement of the optical extinction of the atmospheric aerosol. *Analyst*, 126(8):1216–1220, 2001.

[24] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[25] J. Yan, M. Zhu, H. Liu, and Y. Liu. Visual saliency detection via sparsity pursuit. *IEEE Signal Processing Letters*, 17(8):739–742, 2010.