

Extracting High-level Multimodal Features

Xin Li^{1, a*}, Ruifang Liu^{2, b}

¹Beijing University of Posts and Telecommunications Beijing, Beijing 100876 China

² Beijing University of Posts and Telecommunications Beijing, Beijing 100876 China

^alix@bupt.edu.cn, ^blrf@bupt.edu.cn

Keywords: deep learning; denoising autoencoder, multimodal

Abstract. Consider the problem of building high-level, multimodal features from only unlabeled data, we train model consisting of a sparse stacked denoising autoencoder network with max pooling, which can be used to extract high-level image feature, on a large dataset consisting of multimodal information, and a text treating processes. Our model joints the image feature and text feature as representation of one united movie. We find that these representation can be used in regression mission, predict movie's rating, and the model obtains better effect than unimodal representation.

Introduction

Objects always exists in multimodal ways in real world. For example, a movie consists of not only visual and audio signals, but image information (movie poster) and text information (movie's properties, such as actors and directors). Each modality stands for different properties which help us to classify the object. Multimodal feature can be learned by multimodal signals to capture the object in real world. Different multimodal feature always carries different information. We will just know a word like "sun" but never know the "sun" brings us light and energy until we see it, as a result, only multimodal feature from multimodal learning can extract object in real world.

A useful model should learn representations from data by fusing the modalities into a joint representation that stands for an object in real world. However, only a certain amount of unlabeled data can be obtained practically, so unsupervised learning provides an inexpensive way to learn features. What's more, it answers the question whether substantive features of an object can be learned from those unlabeled data. Comparing to a child's learning procedure, which starts from only unlabeled data, label may not be necessary to a model's learning.

In recent years, unsupervised feature learning and deep learning have been hotly discussed. Effective algorithms and their applications are proposed, like RBMs (Hinton et al., 2006) [1], autoencoders and sparse coding (Olshausen B A, 1996) [2]. Capture complex invariances from simple information is the aim of such algorithms.

In the paper, we build a sparse stacked denoising autoencoder with max pooling to train image data and get multimodal features by jointing both image feature and text feature together. The result satisfies that the similar objects have similar feature, and it can be used in regression mission.

Related Work

Our work is inspired by efficient unsupervised feature learning algorithms and multimodal learning algorithms.

The work of Olshausen & Field in 1996 [2] concluded proposing the sparse coding algorithm. Their work demonstrates the fact that sparse coding work well with unlabeled data. However, the disadvantage of sparse coding is that it belongs to shallow learning rather deep learning, so it can only capture low-level feature. Deep learning, which builds hierarchies feature, can remedy sparse coding's defect. Vincent P, Larochelle H and Lajoie I published their paper in 2010 [3], in which they explore an original strategy for building deep networks, based on stacking layers of denoising autoencoders which are trained locally to denoise corrupted versions of their inputs, and demonstrated the efficiency of the algorithm.

Quoc V. Le and his team considered the problem of building high-level, class-specific feature detectors from only unlabeled data, train a sparse denoising autoencoder with pooling and local contrast normalization, and demonstrate that such detector is sensitive to high-level conceptions, such as cat faces [4]. Nitish Srivastava and Ruslan Salakhutdinov designed a deep Boltzmann machine to extract a unified representation that fuses modalities together and finally result on bi-modal data consisting of images and text. They demonstrated this representation is useful for classification and information retrieval tasks [5].

Model Building

Backpropagation

Backpropagation is an abbreviation for “backward propagation of errors”. It’s always used to train artificial neural networks in conjunction with optimization methods like gradient descent. This algorithm calculates the gradient of a given loss function, and in turn uses the value to update the weights in order to minimize the loss function [6].

The backpropagation learning algorithm consists of two steps: propagation and weight update. The propagation step involves the following steps: forward propagation and backward propagation.

Forward propagation generate the propagation’s output activations by a training pattern’s input through the neural network. Backward propagation of the propagation's output activations through the neural network using the training pattern target in order to generate the deltas, as Eq.1 shows, (the difference between the input and output values) of all output and hidden neurons.

$$J(w) = \frac{1}{2} \sum_{k=1}^d (x_k - z_k)^2 \quad (1)$$

While each weight update follows the next two steps. The first step is Eq. (2).

$$\Delta w_j^{(k)} = -\eta \frac{\partial J(w)}{\partial J(w_j^{(k)})} \quad (2)$$

Multiply output delta and input activation to get the gradient of the weight; subtract a ratio of the gradient from the weight, usually marked as η . This ratio, called the learning rate, influences the speed and quality of learning. The greater the ratio, the faster the neuron trains; the lower the ratio, the more accurate the training is.

The principle of backpropagation is that the sign of the gradient of a weight always indicates where the error is increasing, which is. Backpropagation repeat propagation step and weight update step to train network until the network is satisfactory.

Denoising Autoencoder

An autoencoder neural network is an unsupervised learning algorithm that applies backpropagation, setting the target values to be equal to the inputs [7]. The Fig. 1 shows an autoencoder.

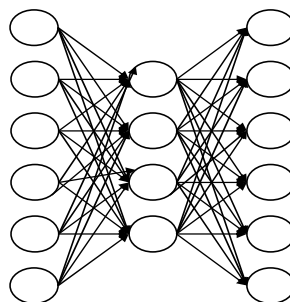


Fig. 1 an autoencoder

Now let the set of unlabeled training examples $\{x_1, x_2, x_3, \dots\}$, where $x_i \in \mathbb{R}$. The hidden layer of autoencoder neural network is $\{y_1, y_2, y_3, \dots\}$, where $y_i \in \mathbb{R}$. The deterministic mapping between inputs and outputs is Eq. (3)

$$y = \sigma(Wx + b) \quad (3)$$

σ is a non-linearity such as the sigmoid. The latent representation y , or code is then mapped back (with a decoder) to a hidden representation y through a deterministic mapping as Eq. (4).

$$z = \sigma(W'y + b') \quad (4)$$

Z , which is given by y , is a prediction of x , and $z = x$. The weight matrix W' of the reverse mapping may be constrained to be the transpose of forward mapping: $W = W'$. The parameters of this model (W, b, b') are optimized such that the average reconstruction error is minimized.

The reconstruction error can be measured in many ways, such as Eq. (5).

$$L(x, z) = -\sum_{k=1}^d (x_k \log z_k + (1 - x_k) \log(1 - z_k)) \quad (5)$$

The denoising auto-encoder is a stochastic version of the auto-encoder. It tries to encode the input and to undo the effect of a corruption process stochastically applied to the input of the auto-encoder. The latter can only be done by capturing the statistical dependencies between the inputs. In order to force the hidden layer to discover more robust features and prevent it from simply learning the identity, a autoencoder is trained to reconstruct the input from a corrupted version of it. The stochastic corruption process randomly sets the specified percentage of the inputs to zero, as shown in Fig. 2.

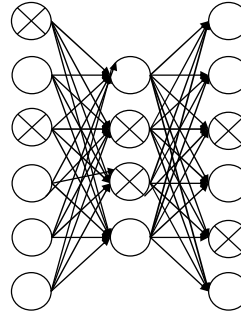


Fig. 2 a denoising autoencoder

Hence the denoising auto-encoder is trying to predict the missing values from the non-missing values, for randomly selected subsets of missing patterns.

Stacked Denoising Autoencoder

In order to form a deep network, stack autoencoders by put the hidden units on the layer below as the inputs of the current denoising autoencoder [3], as shown in the Fig. 3.

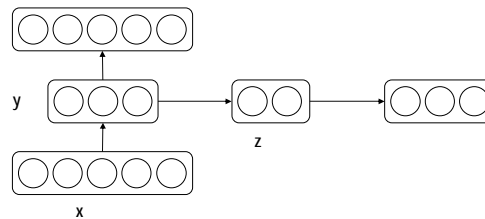


Fig. 3 an stacked denoising autoencoder

let y denote not only the hidden units of the former layer, but also the input units of the next layer. The training process is completed one layer each time. Each layer can be considered as an independent denoising autoencoder and trained by backpropagation algorithm. Once the former layer

are trained, the next layer train could be trained for the latent representation from the former layer has been achieved .

Max Pooling

To address the problem that the input feature of one piece of image is too large. While images have the "stationary" property, which implies that features that are useful in one region are also likely to be useful for other regions. Thus, to describe a large image, one approach is to aggregate statistics of these features at various locations. For example, one could compute the mean (or max) value of a particular feature over a region of the image. These summary statistics are much lower in dimension (compared to using all of the extracted features) and can also improve results (less over-fitting). Such aggregation operation is called this operation pooling, or sometimes mean pooling or max pooling (depending on the pooling operation applied) [8]. The Fig. 4 shows the procedure of max pooling.

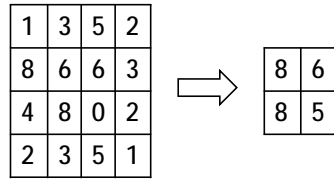


Fig. 4 the procedure of max pooling

Model Description

This section describes our model for building high-level multimodal feature. Consider the image processing part, a denoising autoencoder. Let X denotes the vector of inputs into autoencoder, Y denotes the vector of the hidden layer of autoencoder, and Z denotes the vector of outputs from the hidden layer. W and b is the weight and bias at input layer, while W_prime and b_prime is the weight and bias at hidden layer. The forward propagation is Eq. (6) ~ (7)

$$Y = \sigma(WX + b) \quad (6)$$

$$Z = \sigma(W_prime * y + b_prime) \quad (7)$$

where W_prime is the transpose of W , and σ equals sigmoid function. With denoising, the forward propagation becomes (8) ~ (11)

$$r \sim \text{Bernoulli}(p) \quad (8)$$

$$X = r * X \quad (9)$$

$$Y = \sigma(WX + b) \quad (10)$$

$$Z = \sigma(W_prime * Y + b_prime) \quad (11)$$

Where r is a vector of independent Bernoulli random variables, and each part of r has probability of p of being 1. By multiplying with r , a stochastic version of the autoencoder is constructed to avoid the problem of overfitting.

As shown in Fig. 5, the entire structure of our model is two parts: the first part is used for training

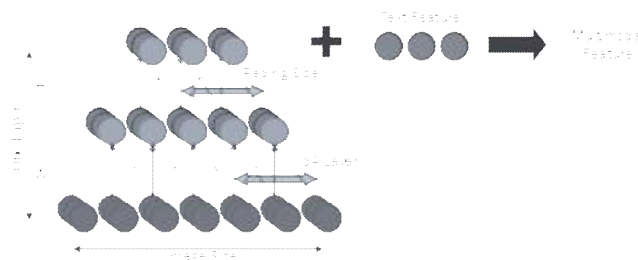



Fig. 5 the structure of the model

image data and gaining abstract image feature; in the second part, we joint image feature with text feature and gain our final result, high-level multimodal feature.

Experiment

Consider the data of inputs, we download image data from the IMDB database, extract movies' properties, such as their actors, directors, plot, and imdbRatings, into a feature vector, and normalize the vector. Table 1 shows the inputs of one movie of the database, 42.

TABLE1 data of inputs

Image	Text
	Title 42
	Genres Biography, Drama, Sport
	Actors Chadwick Boseman, Harrison Ford, Nicole Beharie, Christopher Meloni
	Language English
	Rated PG-13
	Released 12 Apr 2013
	Awards 4 wins & 16 nominations
	...

The image pathway consists of a sparse stacked denoising autoencoder with 1829 visible units followed by two hidden layer of (1000, 500) hidden units and a max-pooling layer of 2*2 units, and each layer is trained in no less than 1000 epochs. The final image vector consists of 125 units. The text pathway consists of a feature vector of 30 units. The multimodal vector consists of 155 units finally. The imdbRating uses ten-point system.

Experiment Results

The first set of our experiments, evaluate the denoising autoencoder and stacked denoising autoencoder as a discriminative model for both multimodal inputs and unimodal inputs.

TABLE 2 RESULTS OF DENOISING AUTOENCODER

	Modality of Inputs	
	<i>Multimodal</i>	<i>Unimodal</i>
Absolute error Mean	0.960	1.047
Absolute error Std	0.988	0.931

As shown in table 1, the denoising autoencoder achieves an absolute error mean of 0.960 for multimodal inputs, which smaller than 1.047 for unimodal inputs.

TABLE 3 RESULTS OF STACKED DENOISING AUTOENCODER

	Modality of Inputs	
	<i>Multimodal</i>	<i>Unimodal</i>
Absolute error Mean	0.915	0.939
Absolute error Std	0.972	0.881

As shown in table 2, the stacked denoising autoencoder achieves an absolute error mean of 0.915 for multimodal inputs, which smaller than 0.939 for unimodal inputs, and its results are both better than denoising autoencoder.

To measure the effect of our model, the left figure in figure (6) shows the distribution of absolute error in denoising autoencoder, while the right figure in figure (6) shows the distribution of absolute error of denoising autoencoder. The x-axis is the data range of absolute error, and the y-axis shows the number of samples the multimodal data subtracts the unimodal subtracts. From those two figures, multimodal learning acquires more valid result than unimodal learning in both the stacked denoising autoencoder and denoising autoencoder.

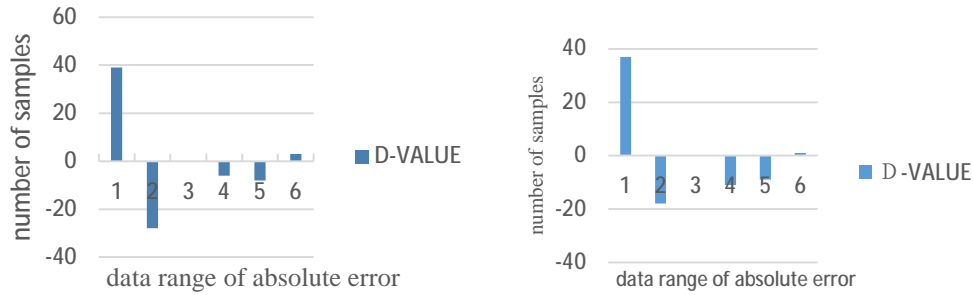


Figure (6) the distribution of absolute error in denoising autoencoder and stacked denoising autoencoder

Conclusion

We propose a model for learning multimodal feature. In this paper, we construct a sparse stacked denoising autoencoder with max pooling to train image feature, fuse multiple data modalities into a unified feature and finally get high-level multimodal feature. Our dataset consisting of only unlabeled data can be effectively used by the model. We use these features in predicting the movies' ratings to confirm its effect. In our experiment, multimodal features runs better compared with unimodal features. And such multimodal features have the same similarity with the objects themselves.

References

- [1] Hinton G E, Salakhutdinov R R. Reducing the dimensionality of data with neural networks[J]. Science, 2006, 313(5786): 504-507.
- [2] Olshausen B A. Emergence of simple-cell receptive field properties by learning a sparse code for natural images[J]. Nature, 1996, 381(6583): 607-609.
- [3] Vincent P, Larochelle H, Lajoie I, et al. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion[J]. The Journal of Machine Learning Research, 2010, 11: 3371-3408.
- [4] Le Q V. Building high-level features using large scale unsupervised learning[C]//Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on. IEEE, 2013: 8595-8598..
- [5] Srivastava N, Salakhutdinov R R. Multimodal learning with deep boltzmann machines[C]//Advances in neural information processing systems. 2012: 2222-2230.
- [6] Rumelhart D E, Hinton G E, Williams R J. Learning representations by back-propagating errors[J]. Cognitive modeling, 1988, 5: 3.
- [7] Vincent P, Larochelle H, Bengio Y, et al. Extracting and composing robust features with denoising autoencoders[C]//Proceedings of the 25th international conference on Machine learning. ACM, 2008: 1096-1103.
- [8] Weng J J, Ahuja N, Huang T S. Learning recognition and segmentation of 3-d objects from 2-d images[C]//Computer Vision, 1993. Proceedings., Fourth International Conference on. IEEE, 1993: 121-128.