Knowledge-Based Systems 102 (2016) 64–75

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Learning robust uniform features for cross-media social data by using cross autoencoders

Quan Guo[a], Jia Jia[b], Guangyao Shen[b], Lei Zhang[a,*], Lianhong Cai[b], Zhang Yi[a]

[a] College of Computer Science, Sichuan University, Chengdu, 610065, China
[b] Department of Computer Science and Technology, Tsinghua University, Beijing, 100084, China

ARTICLE INFO

Article history:
Received 19 November 2015
Revised 25 March 2016
Accepted 26 March 2016
Available online 29 March 2016

Keywords:
Cross-media
Social data
Cross modality
Deep learning
Autoencoder
Convolutional network

ABSTRACT

Cross-media analysis exploits social data with different modalities from multiple sources simultaneously and synergistically to discover knowledge and better understand the world. There are two levels of cross-media social data. One is the *element*, which is made up of text, images, voice, or any combinations of modalities. Elements from the same data source can have different modalities. The other level of cross-media social data is the new notion of *aggregative subject* (AS)— a collection of time-series social elements sharing the same semantics (*i.e.*, a collection of tweets, photos, blogs, and news of emergency events). While traditional feature learning methods focus on dealing with single modality data or data fused across multiple modalities, in this study, we systematically analyze the problem of feature learning for cross-media social data at the previously mentioned two levels. The general purpose is to obtain a robust and uniform representation from the social data in time-series and across different modalities. We propose a novel unsupervised method for cross-modality element-level feature learning called cross autoencoder (CAE). CAE can capture the cross-modality correlations in element samples. Furthermore, we extend it to the AS using the convolutional neural network (CNN), namely convolutional cross autoencoder (CCAE). We use CAEs as filters in the CCAE to handle cross-modality elements and the CNN framework to handle the time sequence and reduce the impact of outliers in AS. We finally apply the proposed method to classification tasks to evaluate the quality of the generated representations against several real-world social media datasets. In terms of accuracy, CAE gets 7.33% and 14.31% overall incremental rates on two element-level datasets. CCAE gets 11.2% and 60.5% overall incremental rates on two AS-level datasets. Experimental results show that the proposed CAE and CCAE work well with all tested classifiers and perform better than several other baseline feature learning methods.

1. Introduction

With the rapid development of the Internet, people have become increasingly dependent on social connections. Social media data are form by multiple modalities, for instance, text, images, voice, social interactions, *etc*. Moreover, the modalities in data samples vary very much. Social media data has created different types of correlational structures and distinctive statistical properties. Traditional approaches focus on dealing with single modality data or fusing data of multiple but same modalities. In contrast, cross-media learning focuses on homogeneous and heterogeneous multimedia data. This multimedia data from various sources needs to be integrated as a means to discover knowledge about the world synergistically. We refer to this problem as the core problem for cross-media learning.

Cross-media social data is based on two levels: the element level and aggregative level. At the element level, users create and spread numerous social media *elements* such as blogs, tweets, photos, and videos across various modalities. These blogs may have photos and videos often contain textual content such as hashtags and title descriptions; however, not every blog has a photo, and not every video includes text. At the aggregative level, collections of cross-media social elements are defined as *aggregative subjects* (AS) by the semantics they share. For example, photos make up an album on image-sharing websites such as Flickr or Instagram, where each album is an AS example; tweets make up a timeline for a user on social networks like Twitter or Facebook, where the timeline is an AS example; questions and comments make up a thread on Q&A communities like StackExchange or Quora where the thread is an AS example. Moreover, during an emergency event

* Corresponding author. Tel.: +862885400618; fax: +862885400618.
E-mail addresses: guoquanscu@gmail.com (Q. Guo), jjia@mail.tsinghua.edu.cn (J. Jia), thusgy2012@gmail.com (G. Shen), leizhang@scu.edu.cn (L. Zhang), clh-dcs@tsinghua.edu.cn (L. Cai), zhangyi@scu.edu.cn (Z. Yi).

http://dx.doi.org/10.1016/j.knosys.2016.03.028
0950-7051/

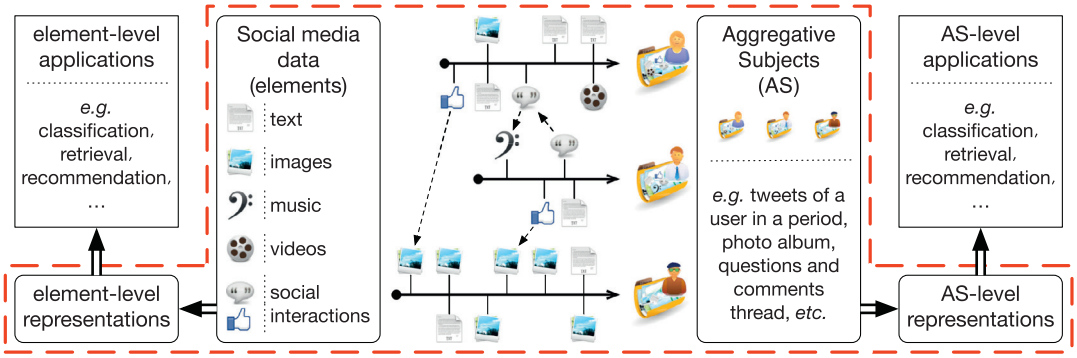

Fig. 1. Illustration of the concepts and applications of learning robust and uniform features for cross-media social elements and AS.

like the Fukushima earthquake, a diverse set of people may upload photos, post tweets, and share blogs about this topic. For example, these social media data about the same Fukushima earthquake topic form an AS sample. Fig. 1 illustrates examples of social elements and AS. There are two characteristics of the elements in an AS: (1) they are in a time-series; (2) each element may contain multiple modalities. Nevertheless, the modalities of the elements may differ from each other. Elements have multiple and different modalities.

Given a social media dataset of elements or AS, the key problem to be addressed is establishing uniform features for unstructured homogeneous and heterogeneous cross-media data. There are certain demands for modeling cross-media social elements and AS for many social media applications including classification, retrieval, and recommendation systems. The representation of the data or the choice of features used to represent the inputs is critical to the overall performance of the applications.

In this study, our goal is to obtain robust features for cross-media social elements and simultaneously extract the uniform features for AS. The problem is non-trivial and poses a set of unique challenges. First, the elements are under a cross-modality setting. They can contain more than one modality. Moreover, their modalities can differ from each other. How do you obtain the modality-invariant representations? Second, the elements in AS are created over time and in time-series. Each of them has a specified context. How do you maximize the use of the time series and context information? Third, there are outliers among the elements of AS. Moreover, there are naturally occurring noise factors among the elements. For example, there can be document images such as “a passport” in a travel album. How to reduce the impact of outliers in data?

The red dashed-line box in Fig. 1 identifies the problem addressed in this study. The solid line with an arrowhead in the red box indicates the timeline of the elements in ASs. In addition, social elements are listed around the timeline. The dashed lines with the arrowheads indicate the targets of social interactions.

Deep learning [1,11,12], utilizing deep architectures and effective learning algorithms, has been emerging as a comprehensive paradigm for a vast range of problems. Krizhevsky et al. demonstrated a considerable improvement on image classification using convolutional neural networks (CNNs) [17]. Deep neural networks also achieve the state-of-the-art in multimedia areas with unstructured data [20,26,33]. Researchers also investigate neural networks for retrieval tasks [7,43]. There are works to integrate deep learning with other intelligence paradigms, for example, Zhou et al. [44] use deep neural networks for a context-aware stereotypical trust model in a multi-agent system.

We formulate the cross-media social elements feature learning problems and AS feature learning problems, respectively. We propose a novel unsupervised method for feature learning of cross-media social elements, namely cross autoencoder (CAE). A two-phase training method for training CAE with massive cross-modality data sample is presented. CAE can learn cross-modality correlation by an inductive cropping strategy, while also making use of the massive data with multiple and different modalities. Furthermore, we propose to use a CNN framework with CAE filters for AS-level feature learning, namely convolutional cross autoencoder (CCAE). We unroll the convolution operation and train CAE filters in CNN offline with the patches extracted from data samples. To the best of our knowledge, CCAE addresses a completely new problem to represent collections of cross-media elements, whereas previous technical works always focus on single independent elements [7,26,33].

Our contributions can be summarized as follows:

- We formulate the feature learning problem for cross-media social data with respect to social elements as well as social AS. We evaluate the quality of the learned features in the context of classification.
- We propose a CAE that learns modality-invariant features from cross-media social elements with different modalities in a two-phase unsupervised manner.
- Applying CAE as filters to handle cross-media elements, we employ a CNN framework to learn features for social AS. The CNN framework can manage the time sequence in social AS and reduce the impact of outliers in the social data.

To evaluate the quality of the proposed learning algorithm, we conduct experiments with classification tasks using real-world datasets from social media websites: Weibo, Sougo, and Flickr. We present experimental results for social elements using CAE and for social AS using CCAE. In terms of accuracy, CAE gets 7.33% and 14.31% overall incremental rates on two element-level datasets. CCAE gets 11.2% and 60.5% overall incremental rates on two AS-level datasets. Results indicate that CAE learns cross-modality correlation from cross-media social data. Further, supervised tasks using features from CAE show significant improvement as compared with baselines, and the experiments for AS show CCAE has superior performance for feature learning.

The remainder of this paper is organized as follows: In Section 2, we formulate the feature learning problem for cross-media social data. In Section 3, we briefly survey some mainstream methods for feature extraction and emphasize deep learning methods using autoencoders. In Section 4, we propose CAE for learning modality-invariant features of social media elements, and CCAE for learning uniform features for AS in social media. In Section 5, we present some experimental results. Section 6 concludes the paper.

Table 1
Multimedia tasks settings comparison. $\{\chi_i\}$ is data with one modality from a heterogeneous data set that D_i and D_k are heterogeneous if and only if $k \neq i$.

Setting / phase	Feature learning	Supervised learning	Testing
Multimedia	$\{\chi_{j,k} \mid j \in D_U\}$	$\{\chi_{j,k} \mid j \in D_V\}$	$\{\chi_{j,k} \mid j \in D_W\}$
Multi-modality fusion	$\{\chi_{j,k} \mid j \in D_U, k \in \Omega\}$	$\{\chi_{j,k} \mid j \in D_V, k \in \Omega\}$	$\{\chi_{j,k} \mid j \in D_W, k \in \Omega\}$
Cross modality	$\bigcup_{k\in\Omega}\{\chi_{j,k} \mid j \in D_U\}$	$\bigcup_{k\in\mathfrak{M}}\{\chi_{j,k} \mid j \in D_V\}, \mathfrak{M} \subset \Omega$	$\bigcup_{k\in\mathfrak{M}}\{\chi_{j,k} \mid j \in D_W\}, \mathfrak{M} \subset \Omega$
Shared representation	$\bigcup_{k\in\Omega}\{\chi_{j,k} \mid j \in D_U\}$	$\bigcup_{k\in\mathfrak{M}}\{\chi_{j,k} \mid j \in D_V\}$	$\bigcup_{k\in\mathfrak{N}}\{\chi_{j,k} \mid j \in D_W\}, \mathfrak{N} \cap \mathfrak{M} = \emptyset$

2. Problem formulation

There are three phases relating to general multimedia tasks: the feature learning phase, the supervised learning phase, and the testing phase. Upon examination of the different data modalities considered, there are several settings pertinent to multimedia tasks: the multimedia setting, the multi-modality fusion setting, the cross-modality setting, and the shared representation setting. The dataset for feature learning, usually large scale and unlabeled, is denoted by D_U; the labeled dataset for supervised learning is denoted by D_V; and the testing dataset is denoted by D_W. We summarize the four settings in Table 1. Traditional research focuses on working with each single multimedia modality. All the labeled and unlabeled training data, as well as the testing data, are within same modality k. Multi-modality fusion considers different modalities as all these are available through all phases. It tries to fuse data from different modalities in either an early [28] or late [5] fusion manner for combining heterogeneous modalities for the same goal. Cross-modality setting aims at learning better representations for modalities with unlabeled data from multiple modalities [25,26]. Data from all modalities is available only during feature learning. Shared representation is more challenging given that the learned features must capture correlations across different modalities to form a modality-invariant representation of data [26]. In the supervised training phase and testing phase, different modalities are presented. Cross-media learning problems can be covered by cross-modality setting and shared representation setting.

Notations that will be used in the rest of the paper are listed in Table 2 for reference.

Definition 1. A social media element, or simply an element, is a multi-modality data sample $\chi_j = [\chi_{j,1}; \chi_{j,2}; \cdots; \chi_{j,K}]$ that has K possible modalities $\chi_{j,k} \in \Re^{M_k}, k = 1, 2, \cdots, K$. χ_j is a $\sum_{k=1}^{K} M_k$ dimensional data vector.

Based on the definition of a social media element, we formulate the general-purpose social media element modeling as a feature learning problem:

Problem 1. General-purpose social media element modeling is a feature learning problem finding a transformation $\mathcal{T}: \chi_j \rightarrow y_j, \forall j$ that maps social media elements $\{\chi_1, \chi_2, \cdots\}$ into their representations $\{y_1, y_2, \cdots\}$. $y_j \in \Re^{M_y}$. The resultant representations $\{y_j\}$ contain significant information about the original elements and can be used as features in other tasks.

AS in social media are collections of social elements. The elements are often organized in time series. The characteristic consistency of elements in AS is the characteristic of the AS item, which is also a base characteristic of each element. However, each element also has its own specific characteristic. And there are noise and outliers. It is more important to learn uniform features for the AS than to consider single elements. The characteristic of AS shall better reflect the facts of the user who post them and supports high-level decision-making. Analysis of any small piece is insufficient for inferring the whole. An AS introduces further difficulties.

Table 2
Table of notations.

Symbol	Description
χ_j	the j-th multi-modality element
$\chi_{j,k}$	the k-th modality in the j-th element
K	the number of possible modalities in elements
M_k	the dimensionality of k-th modality raw feature
y_j	an encoded representation of a social media element
$\mathcal{T}: \chi_j \rightarrow y_j$	the transformation model maps all social media
$\forall j$	elements to their corresponding representations
M_y	the dimensionality of element representations
X_i	the i-th AS sample in a dataset
χ_j^i	the j-th cross-modality element in X_i
N_{X_i}	the number of elements in the i-th AS sample
Y_i	an encoded representation of an AS sample
$\mathcal{T}: X_i \rightarrow Y_i$	the transformation model maps all AS sample to
$\forall i$	their corresponding representations
M_Y	the dimensionality of AS representations
L_i	the classification label of AS sample X_i correlated to representation Y_i
N_l	the number of samples in the training set
N_u	the number of samples in the testing set
x	the input to the autoencoders
y	a representation generated by autoencoders
θ	the autoencoder parameter set $\{W, \hat{W}, b, \hat{b}\}$.
$g; \hat{g}$	the activation functions in neural network
$\hat{x}$	the reconstructions of input x
$J(\cdot)$	the cost function of an autoencoder
Φ	the regularization term
λ	the regularization weight
Ω	the set of indexes of all modalities
$f_k(\cdot)$	CAE encoder on the k-th modality
$\mathfrak{M}$	a non-empty subset of Ω
$\chi_{j,\Omega}$	all the modalities of element χ_j
$\chi_{j,\mathfrak{M}}$	the modalities in $\mathfrak{M}$ of the j-th element
$\hat{\chi}_{j,k}$	the k-th modality of the reconstruction of element χ_j
$\hat{\chi}_{j,\mathfrak{M}}$	the modalities in $\mathfrak{M}$ of the reconstruction of element χ_j
$\hat{\chi}_{j,\Omega}$	all the modalities of reconstruction of element χ_j
$\mathfrak{N}_l$	a proper, non-empty subset of Ω

One major challenge is the integration of the elements in the presence of outliers and high noise. Another challenge comes from the cross-modality nature of social elements. As mentioned before, social elements often have multiple components. There are many off-the-shelf methods designed for dealing with each single modality and integrating multiples of them. It becomes noteworthy that significant modalities may be missing due to factors such as privacy issues. Simply dropping the incomplete elements or ignoring the fact of incompleteness will result in a degraded model and failure to support any reasonable high-level decision.

We formally define the AS in social media as follows:

Definition 2. A sample of AS is a series of multi-modality elements. Let χ_j^i, $j = 1, 2, \cdots, N_{X_i}$ denotes j-th element in the i-th AS sample X_i. The i-th AS sample is $X_i = [\chi_1^i\ \chi_2^i \cdots \chi_{N_{X_i}}^i]$.

We formulate the problem of general-purpose AS modeling as a feature learning problem:

Problem 2. General-purpose AS modeling is a feature learning problem finding a transformation $\mathcal{T}: X_i \rightarrow Y_i, \forall i$ that maps AS samples $\{X_1, X_2, \cdots\}$ into representations $\{Y_1, Y_2, \cdots\}$. $Y_i \in \Re^{M_Y}$. The resultant representations $\{Y_i\}$ can be used as features in supervised learning algorithms.

The performance of cross-media social data feature learning can be evaluated by various tasks such as classification, retrieval, or recommendation. For a clear measurement, we evaluate the quality of the learned features by classification tasks. We formulate the two classification tasks for social media elements and AS as follows:

Task 1. Social media elements classification is to train a classification model from a *training set*$\{(y_{l_1}, L_{l_1}), (y_{l_2}, L_{l_2}), \cdots, (y_{l_{N_l}}, L_{l_{N_l}})\}$ of pairs of elements' representations and labels. Then use the learned model to identify the corresponding labels $\{L_{u_1}, L_{u_2}, \cdots, L_{u_{N_u}}\}$ with representations of the unlabeled social media elements in another *testing set*$\{y_{u_1}, y_{u_2}, \cdots, y_{u_{N_u}}\}$.

Task 2. Social media AS classification is to train a classification model from labeled AS representations *training set*$\{(Y_{l_1}, L_{l_1}), (Y_{l_2}, L_{l_2}), \cdots, (Y_{l_{N_l}}, L_{l_{N_l}})\}$, and then use the learned model to identify the corresponding labels $\{L_{u_1}, L_{u_2}, \cdots, L_{u_{N_u}}\}$ of unlabeled AS samples in another *testing set*$\{Y_{u_1}, Y_{u_2}, \cdots, Y_{u_{N_u}}\}$ of AS representations.

The performance in these classification problems is an evaluation of how feature learning for cross-media social elements and time-series AS can benefit practical applications.

3. Related works

3.1. Feature engineering

From the viewpoint of machine learning, the representation of the data or the choice of features used to represent the inputs is critical to the overall performance [3,4,16,35]. Many previous works have designed useful features manually. The bag-of-words model [9] and lexicons like LIWC2007 [27] and LIWC2007 Simplified Chinese Dictionary [8] are widely used in text modeling. Visual features like scale-invariant feature transform [23] and binary robust invariant scalable keypoints [21] are important features that are used by all types of computer vision tasks. Although designing features manually based on specific domain knowledge has been widely applied, it may be more powerful to learn such features with generic priors [2]. Hand-engineered features are task-specific and difficult to adapt to new tasks or new data domains.

There are also approaches that learn features from data samples. For example, principal component analysis finds the linearly uncorrelated components with largest possible variance, known as principal components [14], in an unsupervised manner. Observations can then be projected to the principal components to get compact representations with the most variance retained. Linear discriminant analysis [18] derives category information by linear analysis. It is a supervised method that finds the direction that maximally separates samples from different categories.

Recent advances in deep learning show the superior ability of deep neural networks to learn features for a vast range of tasks by taking advantage of their deep architecture and a layer-wise unsupervised feature learning phase [1,11,12,16]. The learning model can be either restricted Boltzmann machines (RBM) or autoencoders. The latter are considered the basic blocks of feature learning for deep learning. Researchers have also attempted to use deep neural networks with multiple modalities [7,26,33].

Unlike the previous methods using neural networks for modeling data with multiple modalities [7,26,33,37], an important contribution of CAE that we propose is that it can learn cross-modality correlations from an inductive cropping strategy and also make use of a massive amount of data with multiple and different modalities for feature learning. Furthermore, CCAE focuses on high-level AS features, whereas previous methods consider only low-level independent multimedia data.

3.2. Autoencoders

An autoencoder is a shallow network with only one hidden layer to reconstruct the original input data. The input and output layers are of the same size. The reconstruction can be formulated by

$$\begin{cases} y = g(Wx + b) \\ \hat{x} = \hat{g}(\hat{W}y + \hat{b}), \end{cases} \tag{1}$$

where x is the original input (raw input or stimuli from lower layer); W, $\hat{W}$ and b, $\hat{b}$ are connection weights and bias of encoder and decoder layers, respectively; and $g(\cdot)$ and $\hat{g}(\cdot)$ are nonlinear activation functions of each layer. The sigmoid function is often used as the activation function of deep neural networks.

$$g(z) = \frac{1}{1 + e^{-z}}. \tag{2}$$

y is the representation of x in the hidden layer, while $\hat{x}$ is the reconstruction. Denoting parameters in an autoencoder as $\theta = \{W, \hat{W}, b, \hat{b}\}$, the reconstruction $\hat{x}$ is a deterministic function of x that can be written as $\hat{x}(x; \theta)$. Performance of an autoencoder is measured by a cost function

$$J(x; \theta) = \frac{1}{2}\left\|\hat{x}(x; \theta) - x\right\|^2 + \lambda\Phi(\theta), \tag{3}$$

where the second term $\Phi(\cdot)$ is a regularization term that is often used to induce special characteristics in an autoencoder. After pre-training with an autoencoder in an unsupervised way, the hidden layer learns the statistic of input patterns and represents them with a set of non-linear features. By stacking autoencoders by feeding output representations to a subsequence layer as input, deep features can be produced in a Stacked Autoencoder (SAE).

Vincent et al. proposed a significant extension of the autoencoder based on the idea of making the learned representations robust to partial corruptions of the input pattern [37], namely the Denoising Autoencoder (DAE). During feature learning, each input sample is corrupted in that a fixed number of its components are selected randomly and set to 0. Then, the information of selected components is removed from the particular sample and the DAE is trained to fill up these blanks. The trained model is robust to small irrelevant changes in input.

Recently, researchers have applied deep neural networks to multimedia data. Ngiam et al. propose a bi-modal learning model for integrating audio and video information based on stacked RBM [26]. Srivastava et al. proposed a deep network model that fuses text and image data [33] for classification and information retrieval tasks with remarkable experimental results. Feng et al. applied correspondence autoencoders [7] to the retrieval problem by correlating hidden representations of two uni-modal autoencoders.

Based on autoencoders, our approach is different from previous work in that CAE can learn cross-modality correlations from a massive amount of data with *multiple* and *different* modalities. Cross-media social data come with multiple modalities, most of which are incomplete. Having the capability to learn from multiple modalities and incomplete data is important for the practical social data analysis task. This is what our CAE aiming at and being capable of.

3.3. Convolutional neural etworks

CNNs have a large learning capacity, while having fewer connections and parameters to learn compared with similar sized standard network layers [17,19]. The key idea is to replace the fully connected feed-forward operations between layers by convolution operations. CNNs focus on learning stationary local attributes of images, speech, and other data. We can learn AS-level features from a series of single social media objects and use these features to describe the AS.

A pooling technique is often used in CNNs to reduce the size of feature maps. There are two commonly used pooling operations: max pooling and mean pooling. The former calculates the maximum activation among all activations in the feature map, whereas the latter uses the mean of the activations.

The CCAE proposed in this study is based on the CNN framework. We replace the convolution filters in CNN with CAE to address cross-media data and propose to train the CAE filters offline with patches extracted from that data. Feature learning for cross-media social AS exposes CCAE to a novel and non-trivial problem.

4. The proposed method

As we formulate the problems at two levels, the element level and the AS level, we address the feature learning problem for social data.

In the first problem, we have to deal with cross-media social elements. Cross-modality means that social media elements always contain more than one modality; however, the representations are often non-uniform due to heterogeneous modalities and missing modalities. We propose *cross autoencoders (CAEs)* to learn invariant cross-modality features based on the formulation of autoencoders in Section 3.2.

In the second problem, we learn uniform features for AS, addressing three challenges: time series, cross-modality, and outliers. To this end, we propose a *convolutional cross autoencoder (CCAE)* method. In this method, we employ a CNN framework to manage time series data and avoid outliers. Moreover, we propose to use CAEs as filters in CNN for modality-invariant representation.

The goal of the CCAE is different from models for low-level independent multimedia data, *e.g.*, denoising autoencoder (dAE) [37], the bimodal deep belief network (bDBN) [26], and CAE. CCAE focuses on learning high-level features for AS.

4.1. Learning Social Elements Features with CAE

For a cross-media social element, *i.e.*, data with K modalities, we denote the set of all modality indexes by $\Omega = \{1, 2, \cdots, K\}$. CAE is the leveraging of a set of encoders $\{f_k(\cdot)|k \in \Omega\}$ for each modality and corresponding decoders. Outputs of encoders form a uniform representation. Consider a social element χ_j with modalities $\mathfrak{M}$, which is a subset of the indexes of all modalities that $\mathfrak{M} \subseteq \Omega$. We formulate the leveraging as follows :

$$\begin{cases} f_k(\chi_{j,k}) = w_k \chi_{j,k} \\ y_j = g\left(\sum_{k \in \mathfrak{M}} f_k(\chi_{j,k}) + b\right) \\ \hat{\chi}_j = \hat{g}(\hat{W} y_j + \hat{b}), \end{cases} \tag{4}$$

where w_k is the connection weight for each modality that $W = [w_1, w_2, \cdots, w_K]$. We denote each modality k of the element by $\chi_{j,k}$ and the modalities of the element in the set $\mathfrak{M}$ by $\chi_{j,\mathfrak{M}}$. The corresponding reconstruction is denoted by $\hat{\chi}_{j,k}$ and $\hat{\chi}_{j,\mathfrak{M}}$. Notice $\chi_j = \chi_{j,\mathfrak{M}}$ and $\hat{\chi}_j = \hat{\chi}_{j,\Omega}$. y_j is a uniform representation of χ_j, whereas $\hat{\chi}_j$ is a concatenated reconstruction of inputs from each modality. $\hat{\chi}_j$ is now a deterministic function of $\chi_{j,\mathfrak{M}}$ rather than $\chi_{j,\Omega}$. We can write it as $\hat{\chi}_j(\chi_{j,\mathfrak{M}})$.

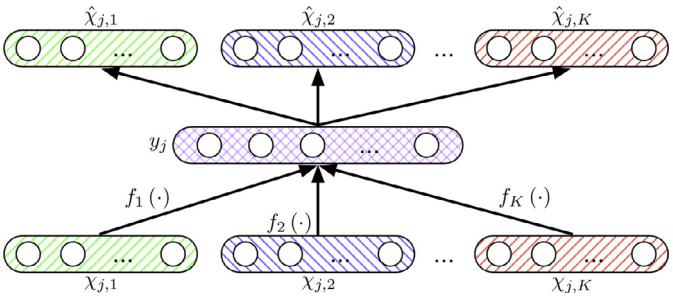

Fig. 2. Illustration of the basic architecture of a CAE.

Fig. 2 depicts the basic architecture of a CAE which receives input from the bottom layer and outputs to the top layer. Cylinders in different colors indicate different modalities. The inputs often include incomplete modalities. The middle layer is the uniform representation of input data. We are looking for a modality-invariant uniform representation in the middle layer. For a specific element χ_j, the representation y_j should be *equivalent* when information from any non-empty set $\mathfrak{M} \neq \emptyset$ of modalities is present. In CAE, we enforce this equivalence making the reconstruction target the same as the input. We require CAE to reconstruct all modalities where only $\mathfrak{M}$ is fed into the encoder, such that $\forall \mathfrak{M} \subseteq \Omega$ and $\mathfrak{M} \neq \emptyset$, $\hat{\chi}_j^i(\chi_{j,\mathfrak{M}}^i) = \chi_j^i$. CAE gets its name from its reconstructing across modalities as well as the cross-modality context it works within.

According to the design of CAE there are two principles: (1) the representations should retain as much information as in the input data, and (2) the representations of input with any combination of modalities should be equivalent. Then we formulate the cost function for CAE as follows:

$$J(\chi_j; \theta) = \frac{1}{2} \| \hat{\chi}_{j,\Omega}(\chi_j; \theta) - \chi_{j,\Omega} \|^2 + \lambda \Phi(\theta). \tag{5}$$

We use the weight decay regularization for generalization in this study. That is, $\Phi(\theta) = \|\theta\|^2$; λ is the penalty weight. We train the autoencoders by minimizing the cost function for all j by stochastic gradient descent.

Another issue we have to deal with is that real world applications always have incomplete data available but not usable due to the limitation of models and training techniques. Fig. 4 shows that labeled data for supervised learning is diminished. Meanwhile, data in the intersection of multiple modalities, which is often required for cross-media learning, is not much.

To maximize the use of training data, we propose a two-phase training strategy: (1) **Augmented Training** (AT) and (2) **Partial Training** (PT). In the first phase, we use data with complete modalities to learn basic cross-modality correlation. In the second phase, we involve all the other data with incomplete modalities.

AT. In the AT phase, we utilize data with all modalities as much as possible. With all modality-complete samples, we construct an augmented sample set. An augmented sample set comprises the input set (feeding to CAE) and the corresponding target set (comparing with decoder output). First, for each combination of modalities, we copy all the modality-complete samples. Second, the modalities that are not in the current combination are set to 0 in this copy. Third, we add this copy to the input set. Correspondingly, we add another copy of all modality-complete samples to the target set. These are repeated for all combinations of modalities. Finally, the CAE is trained by feeding samples from the input set and is required to reconstruct the corresponding samples from the target set.

The procedure to construct the augmented sample set is summarized in Algorithm 1.

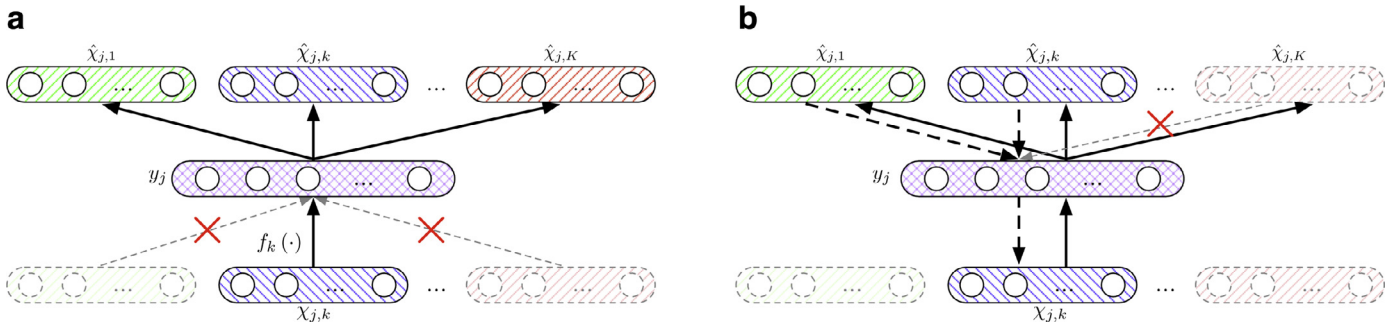

Fig. 3. Illustration of CCAE for an AS sample.

Algorithm 1 Construction of augmented sample set.

Input: dataset with all modalities D; modality combinations configuration $\{\mathfrak{M}\}$

$D_{in} \leftarrow \emptyset$; $D_{tar} \leftarrow \emptyset$
for each combination $\mathfrak{M}$ **do**
  $D_{cropped} \leftarrow D$; $D_{full} \leftarrow D$
  for each modality k that $k \in \Omega$ *and* $k \notin \mathfrak{M}$ **do**
    $\forall \chi_j \in D_{cropped}$, $\chi_{j,k} \leftarrow 0$
  end for
  $D_{in} \leftarrow D_{in} \cup D_{cropped}$
  $D_{tar} \leftarrow D_{tar} \cup D_{full}$
end for

Output: the input set D_{in} and target set D_{tar}

With the augmented sample set, we train the CAE with input from the input set and reconstruction to the target in the output set. Fig. 3a illustrates the feedforward pass of CAE. The lightly colored cylinders with dashed line in the bottom layer indicate the cropped modalities in a sample from the input set. The feedforward passes from these modalities are blocked. For a sample χ_j in the augmented sample set with input modalities $\mathfrak{M}$, the cost function is given by:

$$J(\chi_{j,\mathfrak{M}}, \chi_j; \theta) = \frac{1}{2}\left\| \hat{\chi}_j(\chi_{j,\mathfrak{M}}; \theta) - \chi_j \right\|^2 + \lambda\Phi(\theta). \tag{6}$$

In this phase, CAE learns basic cross-modality correlation among all modalities.

PT. Large amounts of samples in the real world have non-uniform (incomplete) modalities. In PT phase, we make use of samples with incomplete modalities. Because samples of this kind have insufficient information for χ_j in the cost described in Eq. (6), we propose a new partial cost function. We back-propagate the cost from the modalities available in the sample and block from the incomplete ones. As shown in Fig. 3b, the lightly colored cylinders with dashed line in the top layer indicate the incomplete modality in the sample. The backward pass from this modality is blocked. We can categorize samples with respect to the modalities they have. Ignoring $\emptyset$ and Ω, there are $2^K - 2$ possible combination of modalities in this phase. We denote the set of available modalities by $\mathfrak{N}_l \subset \Omega, l = 1, 2, \cdots, 2^K - 2$. The reconstruction of each modality is denoted by $\hat{\chi}_{j,k}$, where $\hat{\chi}_j = \hat{\chi}_{j,\Omega} = \left[\hat{\chi}_{j,1}; \hat{\chi}_{j,2}; \cdots; \hat{\chi}_{j,K}\right]$. Each reconstruction component $\hat{\chi}_{j,k}$ is a deterministic function of $\chi_{j,\mathfrak{N}_l}$, where $\hat{\chi}_{j,k}\left(\chi_{j,\mathfrak{N}_l}\right)$. Then we derive a new cost function as follows:

$$J(\chi_{j,\mathfrak{N}_l}; \theta) = \frac{1}{2}\sum_{k \in \mathfrak{N}_l} \left\| \hat{\chi}_{j,k}\left(\chi_{j,\mathfrak{N}_l}; \theta\right) - \chi_{j,k} \right\|^2 + \lambda\Phi(\theta). \tag{7}$$

In this manner, we ignore reconstruction errors on incomplete modalities because there is no target to compare with. CAE has learned the cross-modality correlation in the first phase with modality-complete samples; thus, we assume that the reconstruction is accurate for the incomplete modalities. This approach allows CAE to use more data samples to obtain a more generalizable model.

An important contribution of CAE is that the proposed method can learn cross-modality correlations from an inductive cropping strategy and also it can make use of massive data collections with multiple and different modalities for feature learning.

The training process of CAE in this study can be viewed as a data augmentation process. Many previous works have suggested that the performance of deep neural networks can be improved by involving more variation in data [31,37]; this is supported by experimental works, such as [17]. State-of-the-art training methods for deep neural networks are optimizing the model locally near the samples [10,16]. One of the practical methods for increasing variation in data is to augment data copies with reasonable data modifications. For example, dAE [37] can be viewed as random data augmentation with pepper noise. Another benefit of data augmentation in our method is that it makes it easier working with skewed data, which is common in social media data. More data from the rare class can be generated. We can apply resample methods and the model can form a robust representation from them. In this way CAE can works well for skewed data.

The idea of augmenting data with respect to the modalities in this study provides a new approach to data augmentation. In the AT phase, data modalities are cropped to form new data samples with same or equivalent semantics in the data space. Seung suggested that an autoencoder, being equivalent to a single iteration of a two layer recurrent network, can learn a low dimensional manifold [32]. In our method, we encourage the CAE to learn a

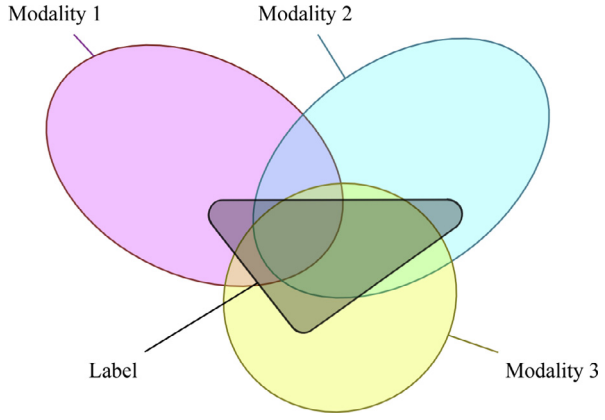

Fig. 4. Data distribution of different modalities and availability of label.

low dimensional manifold in the data space formed by the same sample with the different combination of modalities.

4.2. Learning social AS features with CCAE

To further extend our work to learn uniform features for AS, we use the proposed CAE as modality-invariant filters in a convolution framework. Fig. 5a shows the overall architecture of CCAE. Elements are cylinders listed around the time sequence axis with the time index. Colored sub-cylinders identify different modalities in an element, whereas the empty ones are for incomplete modalities. All such elements form a time sequence, *i.e.*, an AS sample, that is represented as a sequence of cylinders in the figure. Dashed-line cylinders in the middle layer are the local connection patches for CNN. A gray area identifies a typical local connection patch of elements. Rectangles in the upper layer are feature maps using different CAE filters. A gray vertical section identifies CAE filters applying to one patch. Circles on the top are output features.

There are all kinds of sequence problems like path planning [29] and event log mining [36]. In this study, we employ a one-dimensional CNN framework [19,34] to learn AS-level representations. The convolution operation is applied to the elements sequences along a time dimension. To implement this, adjacent elements are connected in a local connection patch. Each filter will be applied on all patches and the activations will form a feature map corresponding to the filter. Then, the feature map will be summarized by a pooling operation. A longitudinal section of the process is shown in Fig. 5b. It shows the connectivity from elements to one output feature. We use these output features corresponding to each filter from the pooling operation as representations of AS samples.

The size of each filter matches the size of a local connection patch. The local connection patch can be view as a cross-modality data item because each element represents cross-modality data. We can treat the local connection patch as one large cross-modality element. Consider, for example, the local connection patch formed by the 2nd, 3rd, and 4th elements in the sequence, as illustrated by Fig. 5c. Fig. 5c is an orthogonal section to Fig, 5b, which contains multiple filters and a single patch. CAE filters receive a patch of samples in the time sequence as the input. The empty circles in the bottom layer indicate the incomplete modalities. Connections indicated by the red dashed lines are blocked. With the modality-invariant uniform representation in the middle layer, CAE reconstructs all modalities including the incomplete ones to the top. The filters used in our model are not manually defined, but trained as CAE. CAE filters receive these inputs from the patches and try to reconstruct all the modalities as mentioned in Section 4.1. Each CAE filter will generate a feature map.

Max pooling and mean pooling are the most popular pooling techniques used in CNNs. Because we pool over time-series elements rather than discrete elements, we consider using mean-over-instances (MOI) and mean-over-time (MOT). For instance, we have eleven tweets collected from seven days from one user, MOI is the total activations divided by eleven (tweets) and MOT is the total activations divided by seven (days). MOI is simply normalized over the total number of activations, whereas MOT is normalized against the time span. In Section 5, we will experimentally evaluate all three operators (MAX, MOI, and MOT) to compare their effects.

Parameters in CCAE are local connection weights and biases of all CAE filters. We learn these parameters with patches from CNN. We extract local connection patches from CNN by unrolling the convolution operation. For convenience, we override the notation χ_j^i for it and use this notation for a local connection patch from now on.

The algorithm to train the CCAE is summarized in the following Algorithm 2.

Algorithm 2 Train a CCAE for cross-modality AS.

Input: AS samples; network setting
1: **for** each sample **do**
2: Unroll convolutional operation and extract local connection patches
3: **end for**
4: $\{\chi_\Omega\} \leftarrow$ patches with all modalities
5: $\left\{\chi_{\mathfrak{N}_l} | l = 1, 2, \cdots, 2^K - 2\right\} \leftarrow$ incomplete patches
6: $\theta \leftarrow$ randomly initialized weights and zero bias
Begin AT phase
7: $D_{in}, D_{tar} \leftarrow$ the input set and target set with $\{\chi_\Omega\}$
8: $\theta \leftarrow$ update weights and bias using the cost in Eq. (6) with inputs in D_{in} and target outputs in D_{tar}
Begin PT phase
9: $\theta \leftarrow$ update weights and bias using the cost in Eq. (7) with all $\chi_{\mathfrak{N}_l}, l = 1, 2, \cdots, 2^K - 2$
Output: parameters θ of the trained network

5. Experimental results and analysis

We present experiments with the proposed methods in two parts. First, we evaluate CAE on social media elements data. Then, we test CCAE against feature learning problems for AS. The experiments are implemented in Matlab[1] with the minFunc[2] optimization toolbox. All the experiments are conducted on a machine with Intel(R) Core(TM) i7-3930K CPU @ 3.20GHz (12 CPUs) and 32 GB RAM.

The performance of feature learning problems for social media can be evaluated on a variety of tasks like retrieval, recommendation, and classification. In this study, we evaluate performance by testing the feature learning methods on classification tasks, using state-of-the-art classifiers. The overall task can be divided into three phases: unsupervised feature learning, supervised learning, and testing. First, we apply comparison methods to learn features from the training set and get representations for all samples. Second, we use the representation obtained from the training set to train a state-of-the-art classification model. Third, we evaluate the classification performance using the testing set.

We measure the classification performance by accuracy and F1-score. Accuracy is the proportion of correct prediction or true results among testing samples. Conversely, F1-score considers both the precision and recall of the prediction. More formally:

$$\begin{aligned} Accuracy &= \frac{TP + TN}{TP + FN + FP + TN}, \\ F_1 &= \frac{2TP}{2TP + FN + FP}, \end{aligned} \tag{8}$$

where *TP*, *FN*, *FP*, and *TN* are the number of True Positive, False Negative, False Positive, and True Negative samples, respectively.

All experiments are conducted with five-fold cross-validation. We randomly divide the dataset into five equal-size parts. For each part, we conduct a set of experiments using it as the testing set and the other four parts as the training set. In this way, we can test all samples in the dataset.

5.1. Experiments with CAE

We evaluate the proposed CAE against two challenging web application tasks: (1) predict stress of users with textual, social

[1] http://www.mathworks.com/products/matlab/.
[2] http://www.di.ens.fr/%7emschmidt/Software/minFunc.html.

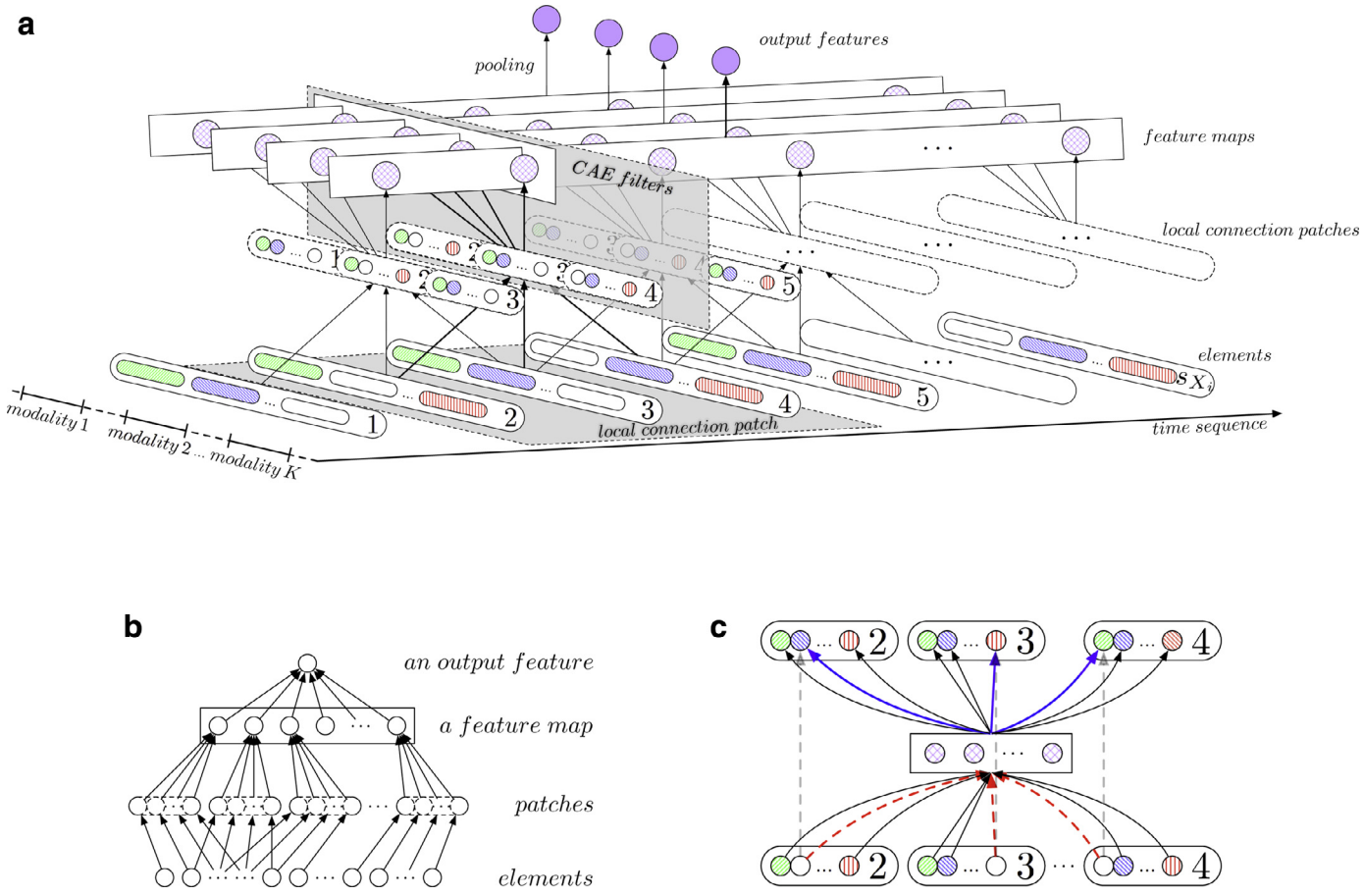

Fig. 5. Illustration of CCAE for an AS sample.

interactive, and visual features from a leading microblog; and (2) predict emotion of users with descriptive, social relevant, and acoustic features from voice assistant application. Both datasets contain features from mixed modalities.

5.1.1. Datasets

Weibo-Stress dataset. Microbloging provides a major platform for people to share their thoughts instantly. As people suffer more and more from the stress of the rapid pace of modern life, a microblog is also a place for people to express their view of life and expose their psychological stress. *Weibo*[3] is the world's largest Chinese microblog website. We use a dataset collected from Weibo with 57,479 items labeled with the psychological lexicon LIWC2007 [27] and LIWC2007 Simplified Chinese Dictionary [8] into 6 categories: Affection, Work, Social, Physiological, Other stress, and None. It comes with ready-made 17-dimensional textual features, three-dimensional social interactive features, and 21-dimensional visual features [39].

Sogou-Emotion dataset. Voice assistants are among the most popular applications on the smartphone. Voice is a natural and direct way to convey one's emotion even without linguistic information. *Sogou*[4] Voice Assistant is among the most well-known voice assistants. We employ a dataset from Sogou Voice Assistant with 238,704 records and related usage information containing the following: descriptive information, time of day, and geometric information. 48,211 records come with six emotion labels: Happy, Sad, Angry, Disgusted, Bored, and Neutral [30]. We use a 70-dimensional descriptive data and a 45-dimensional social feature including time and geometric information as well as a 113-dimensional off-the-shelf acoustic feature [30] in our experiments.

Ground truth. Manually labeling the massive social media datasets for evaluation is time-consuming and impractical. In this place, we follow a compromise method adopted by prior studies on affective computing of social media [40,41]. We first collect the hashtags of social media data provided by users and then extract the affective words among them. The affective word lists related to psychological stress are constructed according to Kamvar et al.'s method [15], and those of emotions are built according to WordNet [6]. The ground truth label for each AS sample is determined by "votes" of the words extracted in its corresponding time interval.

5.1.2. Experimental setup

To evaluate the features with the classification problem, we employed a four-layer network with 400 neurons for each layer and a Softmax output layer for classification. We tested with varying values of parameters $\lambda = 1e-05, 3e-05, 1e-04, 3e-04, 1e-03, 3e-03, 1e-02, 3e-02$. We used three different experimental setups. First, we tested our method in a strictly cross-modality context. We trained CAE in the AT phase with all modalities and used each modality for supervised learning and testing. Second, we incorporated the PT phase and trained our model using data samples with incomplete modalities. Third, we tested the shared representation idea that uses different modalities at the supervised training phase and testing. For a baseline, we use the neural networks with the same physical structure and training parameters as each tested CAE, but train them as standard autoencoders.

[3] http://weibo.com.
[4] http://yy.sogou.com/.

Table 3
Comparison of classification accuracy between baseline and proposed AT in Weibo–Stress dataset. Bold font indicates the best performance for the selected raw features.

	Single modality	Cross-modality
Textual	52.27%	**52.75%**
Social	51.20%	**51.68%**
Visual	78.57%	**79.58%**

Table 4
Comparison between accuracy results using single modality, cross-modality in AT phase, and cross-modality in both AT and PT phase. Bold font indicates the best performance for the dataset.

	Baseline	AT	AT + PT
Weibo–Stress	51.60%	55.38%	**55.68%**
Sogou–Emotion	54.42%	62.21%	**62.86%**

5.1.3. Results and analysis

For the first experiment, we trained CAE in the AT phase using all combinations of data modalities in feature learning. The Weibo–Stress dataset was investigated. After we learned modality-invariant features from the data with all modalities, we stacked up the four-layer network and trained the network with labeled data from each single modality. Testing was conducted with the same modality used in supervised learning. For the baseline, we also tested each single modality using a four-layer network of the same scale.

Comparison results are shown in Table 3. Textual features and social features are weak features compared with visual features. Adding strong features to the feature learning phase of weak features improves the result. Not surprisingly, we obtain the same benefit when weak features are added to feature learning of strong features. The three features get 0.92%, 0.94%, and 1.29% incremental rates, respectively. There are only 3946 items in the Weibo dataset with all three modalities. The experiment is rather limited in that the gain is relatively small. To further take advantage of the large scale of data, we have to introduce the PT phase.

For the next experiment, we try to show the effectiveness of both the AT and PT phases. We created incomplete data from both datasets for the training phase. In the Weibo dataset, we simply use all data. 57,175 items of the data have textual features, 15,744 have the social interactive features while 12,857 have visual features. In the Sogou dataset, we use all unlabeled data and randomly corrupt 60% of the descriptive features, 60% of the social features, and 60% of the acoustic features. There are overlaps of random corruption. Some records may have only one modality left, and some records are even completely corrupted. Completely corrupted records are not used. We thus have 41,087 records with all three modalities, 137,194 records with mixed modalities, and 12,212 abandoned records.

In this second experiment, the supervised learning and testing used data with mixed modalities, unlike the first experiment. For each dataset, we adopted single best modality classifiers as the baseline.

Experimental results are shown in Table 4. Involving multiple modalities with CAE in AT phase gets better result than simply using a single modality model for each sample. We get 7.33% and 14.31% incremental rates for the Weibo–Stress and Sogou–Emotion datasets, respectively. A further increase can be observed when we apply the PT phase to train the model with more incomplete modalities data, where the incremental rates were 7.90% and 15.50%.

For the last experiment of CAE, we demonstrate the capability of shared representation learned by our proposed method. We use the AT and PT phases for feature learning. In supervised learning, we trained the model with the second and third sets of features and tested with the first set of features. That is, for the Weibo-Stress dataset, we trained the model using labeled data with social interactive and visual features and tested with textual data; for the Sogou–Emotion dataset, we trained the model using labeled data with social and acoustic features and tested with descriptive features.

Table 5
Shared representation experiment results compared with random guessing baseline in accuracy. Bold font indicates the best performance for the dataset.

	Baseline	Shared representation
Weibo–Stress	~16.66%	**27.24%**
Sogou–Emotion	~16.66%	**47.17%**

Experimental results are shown in Table 5. It has to be noticed that these results are much better than guessing among six classes by chance ($\sim$ 16.66%). In this experiment, the testing modality has not participated in supervised learning. No explicit information passes between the testing modality and the semantic labels provided. Nevertheless, the model gets significant information from labeled data with the other modalities and the uniform representation. 63.51% and 183.13% incremental rates are achieved for the two datasets respectively. This is evidence that good representation across modalities has been learned.

5.2. Experiments with CCAE

How do we evaluate the quality of the AS-level representations generated by the proposed CCAE? We designed two specialized classification experiments. We first applied comparison methods to learn AS-level representations. Then, we used the representations in the training set to train a state-of-the-art classifier. Finally, we evaluated the classification performance using the testing set. These two classification settings are both about the much-debated topic of affective computing of social media [13,40,41]: predicting users' psychological stress states from the Weibo data and predicting users' emotions from the Flickr data.

5.2.1. Datasets

Weibo-Stress-U dataset. The dataset is also from Weibo, the most popular Chinese microblog platform. It is employed to detect the psychological stress (stressed or not) of a user from his/her tweets over a week. The dataset contains 98,721 tweets from 2843 users over 202 weeks. All tweets have text; there are also images and social interactions. 84,161 (85.25%) tweets have images and 45,708 (46.30%) have social interactions. This is a typical cross-modality setting.

Raw features in this dataset are seven dimensions of text features [42], 21-dimensional color features [24,38,39] and four dimensions of social interaction features [13,41]. In this experiment, tweets are elements, and a user's tweet collection within one continuous week is an AS sample. We extract 9,377 AS samples from the dataset. There are 3106 stressed AS samples and 6271 plain AS samples.

Flickr-Emotion-U dataset. *Flickr*[5] is one of the world's leading image hosting and sharing websites. This dataset is employed to infer emotions (neutral, disgust, happy, or sad) of a photo album's owner over a week with uploaded images. We use a dataset of 177,449 images from 1268 users from January 2008 to March 2013

[5] http://www.flickr.com.

[41]. It is reported that interactions among friends can improve inferring both positive (by 44.6%) and negative (by 60.4%) emotions [41]. In this Flickr dataset, 151,151 (85.18%) of all the images have social interaction.

In this dataset, raw features are 25-dimensional emotion-retaining visual features [24,38,39] and three-dimensional social interaction features [13]. Images and their related properties over a week form an AS sample. Finally, we extract 12,116 AS samples from the dataset. There are approximately 3000 AS samples for each emotion category.

Ground truth. We also use the same method for gathering the ground truth of the data as described in Section 5.1. This time, we collect the hashtags over all elements in the week defined by data.

5.2.2. Experimental setup

Since we formulate the AS modeling problem as an unsupervised feature learning problem, for comparison, we test the proposed method with three other unsupervised feature learning methods:

- Voting: using the classification result of each element to determine the class of AS.
- Principal Component Analysis (PCA) [14]: projecting the observations to principal components of all samples for AS-level representations.
- Convolutional Neural Network (CNN) [19,34]: using standard CNN without the proposed CAE filters.
- Convolutional Cross Autoencoder (CCAE)[6]: using the proposed method.

We considered three classifiers to show the generality of our features: Support Vector Machine (SVM) [3,4]; Random Forest (RF) [22,35]; and Deep Neural Network (DNN) [34,37]. We tuned each classifier with different learned features carefully for a fair comparison.

5.2.3. Results and analysis

Table 6 lists performances of the comparison methods on the two datasets. We highlight the best performance of each tested feature learning method by an underline and that of each classifier by bold font.

Compared to the results of using the element-level voting method, we observe significant improvements by using the AS-level methods (PCA, CNN, CCAE). In terms of accuracy (results by F1 are also shown in Table 6), PCA, CNN, and CCAE gain 7.5% (71.33% over 66.34%), 11.0% (73.64% over 66.34%), and 11.2% (73.79% over 66.34%) incremental rates respectively on the Weibo dataset, and 49.0% (58.93% over 39.56%), 59.3% (63.00% over 39.56%), 60.5% (63.50% over 39.56%) incremental rates respectively on the Flickr dataset. While the voting method is an effective ensemble strategy, it is fragile when there are outliers, which are presented in social media elements. AS representations provide more information for identifying (and avoiding) outliers and providing a global view of all elements. These results reveal that the generated AS-level representations can significantly reduce the impact of outliers.

When compared with PCA, in most cases, CNN and CCAE get better results. In terms of accuracy, on the Weibo dataset, CNN and CCAE have up to 4.6% (73.25% over 70.01%) and 5.0% (73.49% over 70.01%) better results than PCA, respectively. On the Flickr dataset, CNN and CCAE have up to 5.2% (62.00% over 58.93%) and 9.6% (59.81% over 54.57%) better results, respectively. The reason is that the elements of AS in social media are highly time-ordered. The CNN framework can summarize feature maps by pooling methods and forming representations to model time series in AS. Furthermore, CNN framework connects each single element in local connection patches to capture the time-series context, which effectively reduce the impact of outliers.

Finally, CCAE can provide significant gains over CNN for all classifiers. For example, on the Flickr dataset, CCAE shows a 1.6% (60.75% over 59.80%)–9.6% (59.81% over 54.56%) improvement over CNN. Because the social media data are under a typical cross-modality setting, performance will be negatively influenced if cross-modality correlations are not captured. The improvements of the results indicate that the proposed CAE in CCAE can effectively learn modality-invariant features. From the AT and PT phases, CAE gets better generalization ability by enlarging the available data for training.

It takes approximately one hour to train a CCAE with all samples in each dataset on a machine with Intel(R) Core(TM) i7-3930K CPU @ 3.20GHz (12 CPUs) and 32 GB RAM, which is acceptable.

Fig. 6 reveals model performance by different values of three main parameters.

Fig. 6a shows the classification results using different sizes of the augmented sample set in the AT phase. With more augmented data, we obtain better output features and better classification results. However, excessive augmented data increase the variance in data to the model. For balance, we use 250K augmented data samples for optimum performance in our experiments.

Fig. 6b shows the performance with different numbers of output features. A larger number of output features tends to bring better classification performance. However, the improvement becomes inconspicuous after 400 output features. Therefore, we use 400 output features in our experiments, where performance is satisfactory and there is no degradation of the time efficiency of the experiments.

Fig. 6c shows the effect of regularization terms. We tune λ from 3e-5 to 3e-3. With a very small regularization factor, the model can suffer a weak generalization ability, and a very large one will lead to weak features. We select $\boldsymbol{\lambda} = 1e\text{-}4$; this best suits our setting.

5.2.4. Error analysis

We conducted error analysis on the experimental results and observed three major types of source of errors.

First is *abnormal data*. In our datasets, we observed a significant amount of online advertisements (and even spamming) on the social networks. Classifying this type of data composed on purpose is pointless. Filtering such data as spam could reduce the impact to a certain degree, but we still cannot get rid of them.

Second is *empirical raw features*. In our experiments, we use raw features proposed in previous studies. These features are well designed based on linguistic, art, and social observation that simplify classification tasks [8,13,24,27,30,38,39,41,42]. However, learning from raw data may be more powerful in predicting the class label. In addition, we focus on cross-media setting and a uniform representation for AS of social media, not the relevance of input features.

Third is *unreliable labeling*. We label the massive social media dataset following a compromise method adopted by prior studies on affective computing of social media [40,41]. This method heavily depends on hashtags provided by users with the social media elements. However, the hashtags are not always reliable because of mistaken labeling, the habit of abusing hashtags, or intentional irony. These phenomena are not rare on social networks. But such labeling is satisfactory for our evaluation since we have shown the relative improvement over baseline methods on the same dataset.

[6] For CNN and CCAE, local connection size is 3 and stride is 1.

Table 6
Prediction performance (accuracy and F1-score) with Weibo and Flickr data. Underlined numbers indicate the best performance of each feature learning method; bold font indicates the best performance of each classifier.

Data	Classifier	Metric	Voting	PCA - 400	CNN - 400			CCAE - 400		
					MAX	MOI	MOT	MAX	MOI	MOT
Weibo	SVM	Accuracy	66.34%	70.01%	72.93%	73.25%	69.51%	72.97%	**73.49%**	69.32%
		F1-score	46.41%	78.41%	81.45%	81.82%	77.44%	81.41%	**81.90%**	77.23%
	RF	Accuracy	65.61%	71.33%	73.14%	73.64%	70.81%	73.44%	**73.79%**	70.60%
		F1-score	45.07%	79.84%	81.42%	81.49%	79.45%	**81.56%**	81.52%	79.27%
	DNN	Accuracy	65.48%	70.27%	71.94%	73.31%	73.18%	71.64%	73.11%	**73.42%**
		F1-score	45.80%	78.74%	80.96%	81.49%	81.44%	80.74%	81.40%	**81.63%**
Flickr	SVM	Accuracy	34.20%	57.09%	59.21%	59.80%	50.46%	59.62%	**60.75%**	50.35%
		F1-score	26.76%	55.73%	58.35%	58.76%	47.78%	58.72%	**59.83%**	48.18%
	RF	Accuracy	39.56%	54.57%	52.58%	54.56%	51.99%	57.42%	**59.81%**	56.64%
		F1-score	35.49%	54.06%	52.20%	54.13%	51.52%	56.99%	**59.26%**	56.20%
	DNN	Accuracy	34.40%	58.93%	59.10%	62.00%	61.80%	61.93%	**63.50%**	61.97%
		F1-score	28.02%	58.42%	58.75%	61.22%	61.27%	61.13%	**63.01%**	61.22%

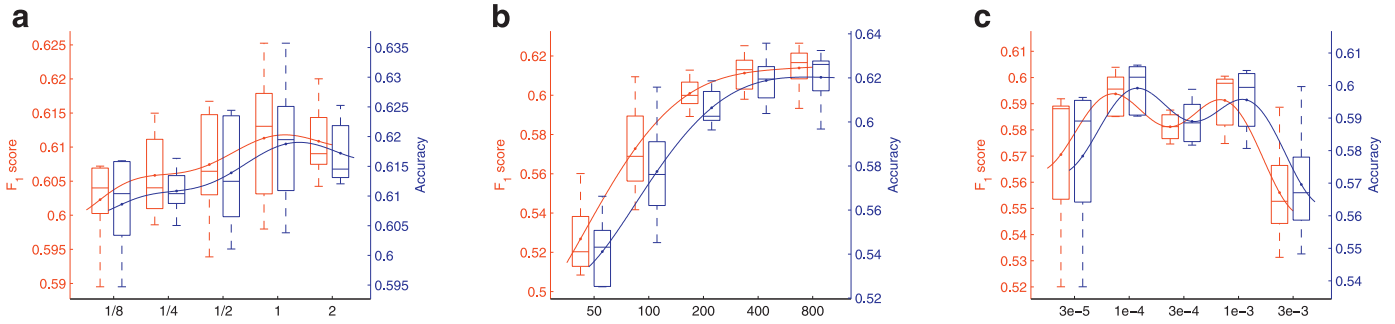

Fig. 6. Classification performance of CCAE features using different parameters. Red curves and boxplots on the left side represent F1-scores; accuracies are shown in blue curves and boxplots on the right side. (a) shows the results using augmented sample sets of different size: 33.75K(1/8), 67.5K(1/4), 125K(1/2), and 250K(1) 500K(2). (b) shows the results using different numbers of output features: 50, 100, 200, 400, and 800. (c) shows the influence of regularization weight *lambda*: 3e-5, 1e-4, 3e-4, 1e-3, 3e-3. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.).

6. Conclusion

People have grown increasingly dependent on online social media interactions. This suggests that it is important to investigate the problem of how to model cross-media social data. We propose two feature learning models to address this problem. To handle the cross-modality correlations in cross-media social elements, we propose CAE to learn uniform modality-invariant features, and we propose AT and PT phases to leverage massive cross-media data samples and train the CAE. To manage the social AS in social media, we further employ a CCAE, which is based on the CNN framework combined with CAE filters. The CNN framework manages the time-series social data, and the CAE filters handle the cross-media social elements. In our approach, the learned AS level features as well as the local connection patches in CNN are much less sensitive to outliers. We present experimental results on several real-world social media datasets to demonstrate that the proposed CAE learns cross-modality correlation in data and CCAE gains significant performance improvements over baseline methods. The proposed CAE and CCAE can be applied to a broad range of social media applications, such as friend recommendations on Weibo and Twitter and photo album retrieval by using keywords.

Acknowledgment

This work is supported by the National Basic Research Program (973 Program) of China under Grant No. 2011CB302201 and Grant No. 2012CB316401, the Key Program of National Natural Science Foundation of China under Grant No. 61432012 and Grant No. U1435213, and also National Natural Science Foundation of China under Grant No. 61370023.

References

[1] Y. Bengio, Learning deep architectures for ai, Found. trends® Mach.Learn. 2 (1) (2009) 1–127.
[2] Y. Bengio, A. Courville, P. Vincent, Representation learning: a review and new perspectives, IEEE Trans. Pattern Anal. Mach. Intell. 35 (8) (2013) 1798–1828.
[3] C.-C. Chang, C.-J. Lin, Libsvm: a library for support vector machines, ACM Trans. Intell. Syst. Technol. (TIST) 2 (3) (2011) 27.
[4] C. Cortes, V. Vapnik, Support-vector networks, Mach. Learn. 20 (3) (1995) 273–297.
[5] H.J. Escalante, C.A. Hérnadez, L.E. Sucar, M. Montes, Late fusion of heterogeneous methods for multimedia image retrieval, in: Proceedings of the 1st ACM international Conference on Multimedia Information Retrieval, ACM, 2008, pp. 172–179.
[6] C. Fellbaum, WordNet, Wiley Online Library, 1998.
[7] F. Feng, X. Wang, R. Li, Cross-modal retrieval with correspondence autoencoder, in: Proceedings of the ACM International Conference on Multimedia, ACM, 2014, pp. 7–16.
[8] R. Gao, B. Hao, H. Li, Y. Gao, T. Zhu, Developing simplified chinese psychological linguistic analysis dictionary for microblog, in: Brain and Health Informatics, Springer, 2013, pp. 359–368.
[9] Z.S. Harris, Distributional structure., Word 10 (2-3) (1954) 146–162.
[10] G.E. Hinton, Training products of experts by minimizing contrastive divergence, Neural. Comput. 14 (8) (2002) 1771–1800.
[11] G.E. Hinton, Learning multiple layers of representation, Trends Cogn. Sci. 11 (10) (2007) 428–434.
[12] G.E. Hinton, R.R. Salakhutdinov, Reducing the dimensionality of data with neural networks, Science 313 (5786) (2006) 504–507.
[13] J. Jia, S. Wu, X. Wang, P. Hu, L. Cai, J. Tang, Can we understand van gogh's mood?: learning to infer affects from images in social networks, in: Proceedings of the 20th ACM International Conference on Multimedia, ACM, 2012, pp. 857–860.
[14] I. Jolliffe, Principal component analysis, Wiley Online Library, 2005.
[15] S.D. Kamvar, J. Harris, We feel fine and searching the emotional web, in: Proceedings of the Fourth ACM International Conference on Web Search and Data Mining, ACM, 2011, pp. 117–126.
[16] H. Kamyshanska, R. Memisevic, The potential energy of an autoencoder, IEEE Trans. Pattern Analy. Mach. Intell. 37 (6) (2015) 1261–1273.

[17] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: F. Pereira, C.J.C. Burges, L. Bottou, K.Q. Weinberger (Eds.), Advances in Neural Information Processing Systems 25, Curran Associates, Inc., 2012, pp. 1097–1105. http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf.
[18] P.A. Lachenbruch, Discriminant analysis, Wiley Online Library, 1975.
[19] Y. LeCun, Y. Bengio, Convolutional networks for images, speech, and time series, The handbook of brain theory and neural networks 3361 (1995).
[20] J. Leng, P. Jiang, A deep learning approach for relationship extraction from interaction context in social manufacturing paradigm, Knowl-Based Syst. (2016).
[21] S. Leutenegger, M. Chli, R.Y. Siegwart, Brisk: binary robust invariant scalable keypoints, in: 2011 IEEE International Conference on Computer Vision (ICCV), IEEE, 2011, pp. 2548–2555.
[22] A. Liaw, M. Wiener, Classification and regression by randomforest, R News 2 (3) (2002) 18–22. http://CRAN.R-project.org/doc/Rnews/.
[23] D.G. Lowe, Object recognition from local scale-invariant features, in: The proceedings of the Seventh IEEE International Conference on Computer vision, 1999, 2, Ieee, 1999, pp. 1150–1157.
[24] J. Machajdik, A. Hanbury, Affective image classification using features inspired by psychology and art theory, in: Proceedings of the International Conference on Multimedia, ACM, 2010, pp. 83–92.
[25] X. Mao, B. Lin, D. Cai, X. He, J. Pei, Parallel field alignment for cross media retrieval, in: Proceedings of the 21st ACM International Conference on Multimedia, ACM, 2013, pp. 897–906.
[26] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, A. Ng, Multimodal deep learning, in: Proceedings of the 28th International Conference on Machine Learning (ICML-11), 2011, pp. 689–696.
[27] J.W. Pennebaker, C.K. Chung, M. Ireland, A. Gonzales, R.J. Booth, The Development and Psychometric Properties of liwc2007, Austin, TX, LIWC. Net, 2007.
[28] T.-T. Pham, N.E. Maillot, J.-H. Lim, J.-P. Chevallet, Latent semantic fusion model for image retrieval and annotation, in: Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management, ACM, 2007, pp. 439–444.
[29] H. Qu, Z. Yi, S.X. Yang, Efficient shortest-path-tree computation in network routing based on pulse-coupled neural networks, IEEE Trans. Cybern. 43 (3) (2013) 995–1010.
[30] Z. Ren, J. Jia, L. Cai, K. Zhang, J. Tang, Learning to infer public emotions from large-scale networked voice data, in: MultiMedia Modeling, Springer, 2014, pp. 327–339.
[31] S. Rifai, P. Vincent, X. Muller, X. Glorot, Y. Bengio, Contractive auto-encoders: explicit invariance during feature extraction, in: Proceedings of the 28th International Conference on Machine Learning (ICML-11), 2011, pp. 833–840.
[32] H.S. Seung, Learning continuous attractors in recurrent networks, in: M.I. Jordan, M.J. Kearns, S.A. Solla. (Eds.), Advances in Neural Information Processing Systems 10, MIT Press, 1998, pp. 654–660. http://papers.nips.cc/paper/1369-learning-continuous-attractors-in-recurrent-networks.pdf.
[33] N. Srivastava, R. Salakhutdinov, Multimodal learning with deep boltzmann machines, Adv. Neural. Inf. Process. Syst. 25 (2012) 2231–2239.
[34] Y. Sun, X. Wang, X. Tang, Deep learning face representation from predicting 10,000 classes, in: Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, IEEE, 2014, pp. 1891–1898.
[35] V. Svetnik, A. Liaw, C. Tong, J.C. Culberson, R.P. Sheridan, B.P. Feuston, Random forest: a classification and regression tool for compound classification and qsar modeling, J Chem Inf Comput Sci 43 (6) (2003) 1947–1958.
[36] J.C. Vidal, B. Vázquez-Barreiros, M. Lama, M. Mucientes, Recompiling learning processes from event logs, Know-Based Syst. (2016).
[37] P. Vincent, H. Larochelle, Y. Bengio, P.-A. Manzagol, Extracting and composing robust features with denoising autoencoders, in: Proceedings of the 25th International Conference on Machine Learning, ACM, 2008, pp. 1096–1103.
[38] X. Wang, J. Jia, L. Cai, Affective image adjustment with a single word, Vis. Comput. 29 (11) (2013) 1121–1133.
[39] X. Wang, J. Jia, J. Yin, L. Cai, Interpretable aesthetic features for affective image classification., in: 20th IEEE International Conference on Image Processing (ICIP), 2013, IEEE, 2013, pp. 3230–3234.
[40] L. Xie, X. He, Picture tags and world knowledge: learning tag relations from visual semantic sources, in: Proceedings of the 21st ACM International Conference on Multimedia, in: MM '13, ACM, New York, NY, USA, 2013, pp. 967–976, doi:10.1145/2502081.2502113.
[41] Y. Yang, J. Jia, S. Zhang, B. Wu, J. Li, J. Tang, How do your friends on social media disclose your emotions? in: Proc. AAAI, 14, 2014, pp. 1–7.
[42] J. Zhao, L. Dong, J. Wu, K. Xu, Moodlens: an emoticon-based sentiment analysis system for chinese tweets, in: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2012, pp. 1528–1531.
[43] G. Zhou, Y. Zhou, T. He, W. Wu, Learning semantic representation with neural networks for community question answering retrieval, Knowl-Based Syst. 93 (2016) 75–83.
[44] P. Zhou, X. Gu, J. Zhang, M. Fei, A priori trust inference with context-aware stereotypical deep learning, Knowl-Based Syst. 88 (2015) 97–106.