Pers Ubiquit Comput
DOI 10.1007/s00779-012-0614-2

ORIGINAL ARTICLE

Self-taught learning via exponential family sparse coding for cost-effective patient thought record categorization

Hua Wang · Heng Huang · Monica Basco · Molly Lopez · Fillia Makedon

Received: 30 September 2011 / Accepted: 21 August 2012

Abstract Automatic patient thought record categorization (TR) is important in cognitive behavior therapy, which is an useful augmentation of standard clinic treatment for major depressive disorder. Because both collecting and labeling TR data are expensive, it is usually cost prohibitive to require a large amount of TR data, as well as their corresponding category labels, to train a classification model with high classification accuracy. Because in practice we only have very limited amount of labeled and unlabeled training TR data, traditional semi-supervised learning methods and transfer learning methods, which are the most commonly used strategies to deal with the lack of training data in statistical learning, cannot work well in the task of automatic TR categorization. To address this challenge, we propose to tackle the TR categorization problem from a new perspective via *self-taught learning*, an emerging technique in machine learning. Self-taught learning is a special type of transfer learning. Instead of requiring labeled data from an auxiliary domain that are relevant to the classification task of interest as in traditional transfer learning methods, it learns the inherent structures of the auxiliary data and does not require their labels. As a result, a classifier achieves decent classification accuracy using the limited amount of labeled TR texts, with the assistance from the large amount of text data obtained from some inexpensive, or even no-cost, resources. That is, a cost-effective TR categorization system can be built that may be particularly useful for diagnosis of patients and training of new therapists. By further taking into account the discrete nature input text data, instead of using the traditional Gaussian sparse coding in self-taught learning, we use exponential family sparse coding to better simulate the distribution of the input data. We apply the proposed method to the task of classifying patient homework texts. Experimental results show the effectiveness of the proposed automatic TR classification framework.

Keywords Major depressive disorder · Cognitive behavior therapy · Thought record · Self-taught learning · Exponential family · Cost-effective classification

H. Wang
Department of Electrical Engineering and Computer Science, Colorado School of Mines, Golden, CO 80401, USA
e-mail: huawang@mines.edu

H. Huang (✉) · F. Makedon
Department of Computer Science and Engineering, University of Texas at Arlington, Arlington, TX 76019, USA
e-mail: heng@uta.edu

F. Makedon
e-mail: makedon@uta.edu

M. Basco
Department of Psychology, University of Texas at Arlington, Arlington, TX 76019, USA
e-mail: basco@uta.edu

M. Lopez
School of Social Work, University of Texas, Austin, TX 78712, USA
e-mail: mlopez@austin.utexas.edu

1 Introduction

In the standard clinic treatment for major depressive disorder (MDD), pharmacotherapy is usually augmented with cognitive behavior therapy (CBT) for those patients who are not responsive to medication alone. CBT is an evidence-based psychotherapy with well-documented efficacy in clinical trials of MDD and is considered as one of the standards used in the field for treatment of depression.

Published online: 30 October 2012

As shown in Fig. 1, a homework in CBT can be described as any activity performed by patients outside the office, which is intended to have a positive effect on therapy. Each homework is assigned with a label corresponding to one of a set of prescribed thought categories. Homework aids patients in generalizing what is discussed during the therapy sessions, promotes learning through practice, and facilitates development of skills such as recognizing negative thoughts. The hallmark of the CBT homework assignment is a thought record (TR). A TR is a means for patients to document their negative automatic thoughts, emotional reactions, and coping behaviors in response to stressful life events. The TR is a critical tool in the therapy process. This analysis and decision-making process is challenging, particularly for new therapists, because it is conducted while the therapist is talking to the patients and attempting to respond in an empathic and therapeutic manner. The process of reviewing and analyzing the TR, however, is very time consuming, which calls for automatic homework processing techniques to facilitate the review and analysis.

Machine learning [2], data mining [21], and information retrieval [16] techniques have provided potential theories and necessary tools to devise new TR analysis methods that are able to automatically process homework in CBT. For example, given the collected homework data in the text as shown in Fig. 1, as well as their associated "thinking error classes" manually labeled by a human expert or a statistical classifier, such as Support Vector Machine (SVM) [3], Naive Bayes (NB) classifier [7], Logistic Regression (LR) [8], Neural Network [1], etc., one can build an automatic system to categorize new coming TRs without additional human intervention.

A potential problem in building an effective automatic TR categorization system is the lack of training data, including the TR data and their corresponding category labels. Such a situation prevents one from training an effective classification model, which usually requires a large number of labeled TRs. Statistically speaking, the more labeled training data one can obtain, the more accurate the classification model can be trained. Because the process of collecting CBT homework results is costly, thought records are not easy to obtain. Moreover, because thought records are collected from people coming from a variety of regions with different languages and habitual traits, analyzing the data is difficult, requires the expertise of trained human labelers, and makes the data labeling process expensive. As a result, *cost-effective classification* methods that do not rely on a large amount of (labeled) TR records are desired, so that data collection costs can be lowered and human labeling efforts can be reduced. In this paper, we explore this challenging, yet important, diagnostic content analysis problem by approaching it from a new perspective using *self-taught learning*, a special type of transfer learning.

1.1 Self-taught learning for cost-effective classification

A straightforward remedy to deal with the lack of training data is to exploit unlabeled data by using semi-supervised

Fig. 1 Example homework texts in CBT

Automatic Thought	Thinking Error General Category	Specific Thinking Errors
She's mad at me for some reason	2	3
And she's trying to get me	2	3
I still suffer from depression and people don't understand that	2	3
My dad, even my dad thinks I'm being lazy	2	3
But people don't understand, man.	2	3

(a) Sample thought records.

General categories	Specific categories
1. MISPERCEPTIONS	1. Magnification 2. Minimization
2. MAKING GUESSES	3. Mindreading 4. Fortune Telling 5. Personalization 6. Overgeneralization 7. Emotional reasoning
3. TUNNEL VISION	8. Mental filtering
4. ABSOLUTES	9. Black and White Thinking 10. Labeling 11. Shoulds and musts

(b) Thinking error types.

learning algorithms [27]. This assumes that the unlabeled data are drawn from the same distribution as those labeled and that their labels are merely unobserved. These assumptions, however, often cannot be satisfied in real-world applications. For example, for the task of TR categorization, a very limited amount of labeled and unlabeled TR data is available. Any assistive text data that could help the classification cannot be considered to be homogeneous, i.e., drawn from the same distribution as the TR text data.

Another broadly used strategy in machine learning and data mining to tackle the training data deficiency is to employ transfer learning algorithms [17, 25, 26]. This strategy is able to discover useful representations from labeled data coming from different distributions. However, typical transfer learning methods require labeled data from a different but related task. This means knowledge is transfered from one supervised learning task to another. Thus, transfer learning requires additional labeled data, rather than unlabeled data, for these other supervised learning tasks, which of course may be expensive to obtain in many applications. Because TR categorization is a data analysis task for very specific cognitive data, very limited comparable text data can be found from inexpensive resources.

Although TR data are rare, we have an overwhelming amount of free texts available from a variety of affordable resources, such as newspapers, the internet, and many other sources. As a result, classification techniques that are able to exploit these inexpensive, or even no-cost, free text data to boost TR categorization tasks may be of practical use for diagnosis.

In this work, we ask how unlabeled text data for other classes—which are much easier to obtain than text data specifically labeled for some certain classes such as thinking error classes in CBT—can be used. For example, given unlimited access to unlabeled and randomly chosen text data, e.g., those downloaded from the internet (probably none of which contains information related to the homework in CBT), can we do better on the given supervised automatic TR classification task?

Motivated by the observation [18, 19] that even many randomly downloaded text will contain the basic semantic patterns that are similar to those in texts of our interested think error classes, we consider to learn a succinct and higher-level feature representation of the inputs using unlabeled data, which could make the classification task of interest easier. From machine learning perspective, our approach belongs to an emerging topic of *self-taught learning* [4, 18, 19], a special type of transfer learning. Because self-taught learning places significantly fewer restrictions on unlabeled data, it is much easier to apply it in TR classification than typical semi-supervised learning or transfer learning methods. For example, it is far easier to obtain 10,000 paragraphs of free texts from the internet than to obtain 10,000 paragraphs of TR records, not to mention 10,000 paragraphs of labeled TR records.

1.2 Self-taught learning via exponential family sparse coding

An important assumption of traditional sparse coding, which lies in the core of self-taught learning, is that the input data are continuous and come from a Gaussian distribution. The Gaussian distribution is applicable to a large number of real-world applications. However, due to the nature of data abstraction of text inputs, where we count the number of the appearances of a set of keywords, the input data are discrete (see Sect. 2.2). As a result, the Gaussian distribution is not able to accurately simulate the input text data for TR categorization. To tackle this, we relax the Gaussian assumption on the data distribution by placing sparse coding under a probabilistic framework [13, 14]. Specifically, we assume the input data come from a member distribution of the exponential family, which could fit discrete input data better, such as the binomial distribution or Poisson distribution. Due to the convexity of the natural parameter function of exponential family distributions, the optimization objective is convex with respect to each of its variables, which makes the solution algorithms computationally tractable. In order to address text input data for TR categorization, instead of using the original self-taught learning approach [4, 18, 19], we use self-taught learning via exponential family sparse coding [13] to achieve better TR categorization results.

1.3 Our classification model via self-taught learning

Under the framework of transfer learning, we have two separate data sets, one from the *auxiliary domain* (also called as *source domain* in some research papers) where we have easy access to abundant, affordable, or even no-cost text data, and the other from the target domain in which the (labeled) text are expensive to obtain. In the scenario of automatic TR categorization, the former corresponds to a resource, such as the internet, where we can obtain unlimited free texts, and the latter corresponds to the TR records collected for diagnosis. Our goal is to build a classifier that is able to accurately classify the text data in the target domain, with the assistance from the texts in the auxiliary domain. Because both the collection and labeling of TR texts are costly, instead of requiring a large of amount expensive labeled TR data as for traditional statistical classification models, we aim to learn a classifier using a small amount of TR data with the help of a large amount of free text data, so that the training cost of our model is comparably lower. We therefore call our TR

classification model a *cost-effective patient TR categorization system*.

We consider to use self-taught learning, an emerging technique of transfer learning, where we do not require the data in the auxiliary domain to be labeled. In the settings of self-taught learning, we first gather a collection of texts with a sufficiently large size from some free resources, such as newswires that are publicly available through the internet, from which we learn a set of semantic prototypes. Because the size of the collected free text is sufficiently large, the variances captured by the learned prototypes are rich enough to cover most, if not all, semantic traits of a certain human language. Then we represent the TR texts to be categorized in the target domain in a new way with respect to the learned semantic prototype set. Because the learned semantic prototypes are learned from a huge collection of texts, the new representation of the target texts is more semantically meaningful than its original counterpart in the form of raw human language representations. As a result, although the data, as well as their labeling information, in the target domain are limited, the learned classifier achieves satisfactory accuracy by leveraging the vast amount of information from general human semantic representations. Namely, the training cost of an effective automatic TR categorization system is decreased. The whole TR categorization system via self-taught learning is schematically shown in Fig. 2. We will detail each component of the proposed system in the next section.

This work expands our previous conference publication [24]. In this manuscript, instead of assuming the input data to be continuous and drawn from a Gaussian distribution, we allow the input data to be discrete, which is closer to the results of the data abstraction model for text inputs. Specifically, when we compute the high-level data representations in self-taught learning, we perform sparse coding using the member distributions in the exponential family other than the Gaussian distribution. Because the data fits better, the classification accuracy in TR categorization is improved.

2 Self-taught learning for TR categorization

In this section, we first formulate the problem of self-taught learning in the scenario of automatic TR categorization and introduce frequently used notation. We then describe the components of the proposed TR classification system one by one in detail.

2.1 Problem formalization of self-taught learning

In self-taught learning [18, 19], a labeled training set $\{(\mathbf{x}_i^l, \mathbf{y}_i^l)\}_{i=1}^n$ is given, drawn independently and identically distributed (i.i.d.) from some distribution $\mathcal{D}$. Each $\mathbf{x}_i^l \in \mathbb{R}^p$ is an input feature vector, and $\mathbf{y}_i^l \in \{0,1\}^K$ is its binary label indication vector for the K concerned classes such that $\mathbf{y}_i^l(k) = 1$ if $\mathbf{x}_i^l$ belongs to the kth class, and 0 otherwise. A set of m unlabeled texts $\{\boldsymbol{x}_i^{\boldsymbol{u}} \in \mathbb{R}^{\boldsymbol{p}}\}_{\boldsymbol{i=1}}^{\boldsymbol{m}}$ is also given. We *do not* assume that the unlabeled texts $\mathbf{x}_i^u$ $(1 \le i \le m)$ were drawn from the same distribution as the labeled data, *nor* do we assume that they can be associated with the same class labels as the labeled data. Our goal is to learn from the labeled data set $\{\mathbf{x}_i^l, \mathbf{y}_i^l\}_{i=1}^n$ as well as the unlabeled data set $\{\mathbf{x}_i^u\}_{i=1}^m$ a function that is able to predict labels of an unseen TR record x under the distribution of $\mathcal{D}$.

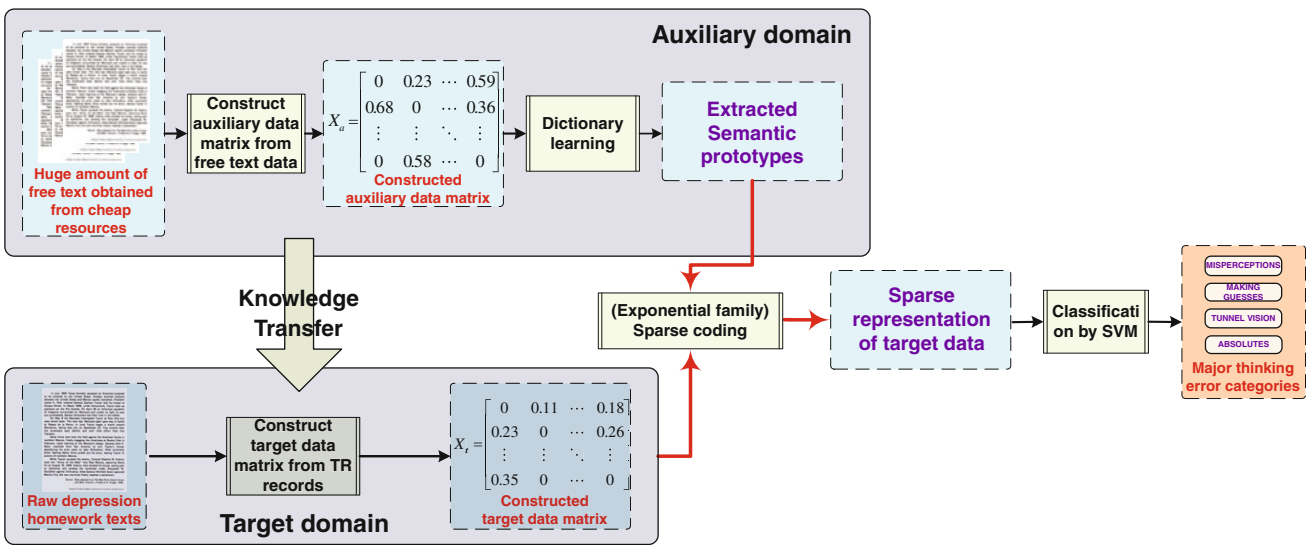


Fig. 2 Diagram of the proposed automatic cost-effective patient thought record categorization system via *self-taught learning* method

Throughout this paper, we write matrices as bold uppercase letters and vectors as bold lowercase letters.

2.2 Construction of input data

Given the raw text data in both the auxiliary domain and the target domain, we first need to abstract them into mathematical forms that can be readily fed into the classification models. Because each depression homework in CBT appears as a short passage of text, we employ the *bag-of-word (BOW)* model [16], a broadly used model in text mining and information retrieval, to construct input data. This consists of two major steps, the dictionary construction and the data representation.

First, we construct a dictionary in which each element (term) is used as a *feature* in the subsequent data representation process. Specifically, given a collection of homework texts, we use all the involved words as terms of the dictionary, from which standard English stop words[1] are removed.

Second, given the constructed dictionary, we represent each homework text using a *tf-idf* weight (term frequency-inverse document frequency) [16]. To be more concise, we assign to each term t a weight with respect to each homework d as the number $n_{t,d}$ of the occurrences of t in d. Taking into account the varied length of the homework texts, we normalize $n_{t,d}$ by the total number of words appearing in the homework d and define its *term frequency (tf)* as:

$$\mathbf{tf}_{t,d} = \frac{n_{t,d}}{\sum_t n_{t,d}}. \tag{1}$$

The raw term frequency defined in Eq. (1) suffers from a critical problem in that all terms are considered equally important. However, certain terms have little or no discriminating power in determining relevance, e.g., a term appearing in all the homework texts. To address this problem, we introduce a mechanism for attenuating the effect of terms that occur too often in the collection to be meaningful for relevance determination. We define the *document frequency* $\mathbf{df}_t$ as the number of documents in the collection that contain a term t [16]. Given the total number N of documents in a collection, we define the *inverse document frequency (idf)* as follows:

$$\mathbf{idf}_t = \log \frac{N}{\mathbf{df}_t}. \tag{2}$$

Finally, each entry x_{ij} of the input data $X \in \mathbb{R}^d \times n$ for the n homework texts for the d terms (dimensions) of the dictionary is computed as

$$x_{ij} = \mathbf{tf}_{i,j} \times \mathbf{idf}_i. \tag{3}$$

[1] http://www.textfixer.com/resources/common-english-words.txt.

2.3 Self-taught learning to represent TR records with enriched semantic prototypes

Given the abstracted data in the auxiliary domain $\{\mathbf{x}_i^u\}_{i=1}^m$ and those in the target domain $\{\mathbf{x}_i^l, \mathbf{y}_i^l\}_{i=1}^n$, we aim to learn a semantical meaningful and discriminative representation of the target data.

In self-taught learning [19], a set of r basis vectors, $\{\mathbf{d}_j \in \mathbb{R}^p\}_{j=1}^r$, is first learned that form a semantic prototype set $\mathbf{D} = [\mathrm{d}_1, \ldots, \mathrm{d}_r] \in \mathbb{R}^{p \times r}$ (allowing $r > p$ to make the prototype set overcomplete), from unlabeled data by minimizing the following objective:

$$\begin{aligned} J_u(\mathbf{D}, \mathbf{a}_i^u) &= \sum_{i=1}^{m} \left(\|\mathbf{x}_i^u - \mathbf{D}\mathbf{a}_i^u\|_2^2 + \lambda \|\mathbf{a}_i^u\|_1 \right), \\ s.t.\ &\|\mathbf{d}_j\|_2 \le 1,\ \forall\ 1 \le j \le r, \end{aligned} \tag{4}$$

where $\lambda > 0$ is a parameter and $\mathbf{a}_i^u \in \mathbb{R}^r$ is the representation coefficient vector of $\mathbf{x}_i^u$ with respect to dictionary $\mathbf{D}$. Here the constraints on $\mathbf{d}_j$ are used to avoid a degenerate solution—the reconstruction errors in the first term of J_u are invariant to scaling simultaneously $\mathbf{D}$ by a scalar and $\mathbf{a}_i^u$ by its inverse. Because of the ℓ_1-norm regularization on $\mathbf{a}_i^u$, it is sparse with very few non-zero entries [22]. $\mathbf{d}_j (1 \le j \le r)$ are usually considered as high-level feature prototypes of the unlabeled data and convey more semantic information [15, 19]. Here, we solve the optimization problem in Eq. (4) using the software package published in [12].[2] Then, we represent the labeled TR records, as well as the unseen TR records, with respect to the learned semantic prototype set $\mathbf{D}$ by minimizing the following optimization objective:

$$J_l(\mathbf{a}_i^l) = \|\mathbf{x}_i^l - \mathbf{D}\mathbf{a}_i^l\|_2^2 + \lambda \|\mathbf{a}_i^l\|_1,\ \forall\ 1 \le i \le n \tag{5}$$

where $\mathbf{a}_i^l \in \mathbb{R}^r$ is the new representation of $\mathbf{x}_i^l$ with respect to $\mathbf{D}$. Again, $\mathbf{a}_i^l$ is sparse due to the ℓ_1-norm regularization. We use the same software package as before to solve the optimization problem in Eq. (5).

2.4 Classification of unseen TR records

Given the learned new representations of the labeled TR records $\{\mathbf{a}_i^l\}_{i=1}^n$, we learn a SVM [19] to classify unseen TR records under the same distribution $\mathcal{D}$ from which $\{\mathbf{x}_i^l\}_{i=1}^n$ were drawn, where the unseen TR records are also represented with respect to the learned semantic prototype set following the same way as $\{\mathbf{a}_i^l\}_{i=1}^n$ by solving Eq. (5).

In the task of TR categorization, we consider each type of thinking error as a class and conduct classification on the

[2] http://www.eecs.umich.edu/honglak/softwares/nips06-sparsecoding.html.

Table 1 Thinking error classes used in this paper

Category	Detailed description
MISPERCEPTIONS	Seeing things as much greater or much smaller than they really are
MAKING GUESSES	Making guesses or jumping to conclusions that are overly negative
TUNNEL VISION	Seeing only the things that confirm your negative view while ignoring positive experiences
ABSOLUTES	Overly harsh, perfectionistic, or strict ideas or statements about how things are or ought to be

participants. Specifically, we consider the four categories of major thinking error listed in Table 1 as four different classification tasks.

The standard SVM algorithm deals with a binary classification problem that comprises only two classes, the positive class and the negative class. The label indicator is $y_i = \{-1, +1\}^n$, such that $y_i = +1$ if data point $\mathbf{x}_i$ belongs to the positive class, and -1 otherwise. The standard SVM algorithm optimizes a linear classifier $\mathbf{w}$ so that the margin between the data points of the two classes is maximized [3]. The learning problem can thus be interpreted as the problem of solving a quadratic constrained optimization problem whose primal form is

$$\begin{array}{ll} \min\limits_{\mathbf{w},\xi,b} & \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum\limits_{i=1}^{n}\xi_i, \\ s.t. & y_(i)(\mathbf{w}^T\phi(\mathbf{x}_i) + b) \geq 1 - \xi_i, \\ & \xi_i \geq 0, \quad 1 \leq i \leq n, \end{array} \tag{6}$$

and whose dual form is

$$\begin{array}{ll} \max & \sum\limits_i \alpha_i - \frac{1}{2}\sum\limits_{i,j}\alpha_i\alpha_j y_i y_j \mathcal{K}(\mathbf{x}_i, \mathbf{x}_j), \\ s.t. & 0 \leq \alpha_i \leq C, \\ & \sum\limits_i \alpha_i y_i = 0, \quad 1 \leq i \leq n, \end{array} \tag{7}$$

where b is a threshold, the ξ_i are slack variables necessary for the case when the training data points are not linearly separable, and C is the error penalty. $\mathcal{K}(\mathbf{x}_i, \mathbf{x}_j) = \langle\phi(\mathbf{x}_i), \phi(\mathbf{x}_i)\rangle$ is a kernel function, by which a data point $\mathbf{x}_i$ is mapped into a higher (maybe infinite) dimensional space by the mapping function ϕ. Thus, the class membership assigned to an unseen data point $\mathbf{x}$ is given by

$$\text{sign}\left(f(\mathbf{x}) = \sum_{i=1}^{n} y_i\alpha_i\mathcal{K}(\mathbf{x}_i, \mathbf{x}) + b\right), \tag{8}$$

where $f(\mathbf{x})$ is the decision function of the SVM.

Extending SVM classification to multi-class classification has been well studied and many approaches have been devised [5], among which the "one-against-one" approach [11] is widely used because of its advantages [9]. In this approach, $K(K-1)/2$ classifiers are constructed and one for each pair of two different classes. After that, a voting strategy is used: each binary classification is considered to be a voting, and a data point is finally designated to be in the class with a maximum number of votes.

3 Self-taught learning via exponential family sparse coding for discrete inputs

In this section, following Lee et al. [13], we first analyze sparse coding from a probabilistic perspective. We derive the method of "exponential family sparse coding" for self-taught learning, which is suitable for discrete input data, such as the text data used for TR categorization.

3.1 A probabilistic interpretation of sparse coding

As in the first term of Eqs. (4) and (5), self-taught learning essentially learns a set of high-level semantic patterns $\mathbf{D}$ and represents the input data in the target domain, as well as those in the source domain, by the linear combinations of the learned semantic patterns. Specifically, given a data point $\mathbf{x}_i$ from either the auxiliary domain or the target domain,[3] we approximate it as

$$\mathbf{x}_i \approx \sum_j \mathbf{d}_j[j]a_{ij}, \tag{9}$$

where a_{ij} is the jth entry of $\mathbf{a}_i$. Hence $\{\mathbf{a}_i\}$ are called as *activations* corresponding to $\mathbf{x}_i$. Due to the imposed ℓ_1-norm penalty on $\mathbf{a}_i$ in the second term of Eqs. (4) and (5), $\mathbf{a}_i$ is encouraged to be sparse [22], i.e., each $\mathbf{x}_i$ is represented by a small number of basis vectors in the learned dictionary $\mathbf{D}$.

In traditional sparse coding [13], the input data vectors are assumed to be continuous and generated from a Gaussian distribution with mean

$$\eta = \sum_j \mathbf{d}_j a_{ij} \tag{10}$$

and (known) covariance matrix $\sigma^2\,\mathbf{I}$ [13, 14]. In addition, the activation vector $\mathbf{a}_i = [a_{i1}, \ldots, a_{ir}]$ is assumed to follow a Laplacian prior

$$P(\mathbf{a}_i) \propto \prod_j \exp(-\beta|a_{ij}|) \tag{11}$$

for some constant β [6, 13, 20]. Then, given the input data $\{\mathbf{x}_i\}_{i=1}^m$, the maximum-a-posteriori (MAP) estimate of the corresponding activations $\{\mathbf{a}i\}_{i=1}^m$ can be obtained by solving [13]:

[3] Here we drop the superscript "l" and "u" for brevity, as there is no difference between auxiliary data and target data when discussing sparse coding.

$$\max_{\{\mathbf{d}_i\},\{\mathbf{a}_i\}} \prod_i P(\mathbf{x}_i|\{\mathbf{d}_i\},\{\mathbf{a}_i\})P\{\mathbf{a}_i\}. \quad (12)$$

Taking the negative logarithm of Eq. (12), we can recover Eq. (4), where constraints on $\mathbf{d}_j$ are added to avoid a degenerate solution, as suggested by Lee et al. [12].

An important insight is revealed by Eq. (12), which lies in the Gaussian distribution assumption on input data of the traditional sparse coding method. Although the Gaussian distribution is the most widely used probabilistic model in statistically learning, it is not universally applicable to any input data in real applications, especially for discrete input data such as text data used in TR categorization.

In the simplest bag-of-words model for text abstraction, each text is described as a fixed length binary vector, in which "1" indicates the presence of a corresponding keyword, while "0" indicates the absence. Traditional Gaussian sparse coding assumes each entry of the input vector is continuous and decomposes the input data and approximates it by $\tilde{\mathbf{x}}_i = \sum_j \mathbf{d}_j[j]a_{ij}$. This leads to the unconstrained sum $\sum_j \mathbf{d}_j[j]a_{ij}$, which is obviously a poor simulation of binary data. Instead, if one could find an approximation of the form $\tilde{\mathbf{x}}_i = \sigma(\sum_j \mathbf{d}_j[j]a_{ij})$, where $\sigma(v) = \left[\frac{1}{1+e^{-v_1}}, \frac{1}{1+e^{-v_2}}, \ldots\right]$ represents the element-wise logistic function for a vector v. Because the logistic function always lies in (0, 1), this new approximation may better fit the input data. As a result, one can replace the Gaussian assumption on the input data by the binomial assumption to use logistic loss in sparse coding to get better data representations. To be more general, one can use any probabilistic model in sparse coding model in Eq. (12) upon the prior knowledge of the input data, such as inputs consisting of nonnegative integer counts of keywords in the form of $\mathbf{x}_i = [0, 1, 2, \ldots]^k$. Due to the convexity property of the natural parameter function, the exponential family distributions have demonstrated the best utility in sparse coding, as well as self-taught learning [13].

The remainder of this section will introduce "exponential family sparse coding" and apply it to TR categorization, which was originally proposed by Lee et al. [13].

3.2 Exponential family sparse coding for discrete inputs

The exponential family is a widely used class of distributions in statistics, which can be written in its most general form as [13]:

$$P(\mathbf{x}_i|\eta) = h(\mathbf{x}_i)\exp(\eta^T T(\mathbf{x}_i) - \alpha(\eta)), \quad (13)$$

where η is the natural parameter for the model, and the functions h, T and α together define a particular member of the family. It can be verified that the multivariate Gaussian distribution used in traditional sparse coding can written in the form of Eq. (13) with $h(\mathbf{x}_i) = e^{-\frac{\|\mathbf{x}_i\|^2}{2}}/(2\pi)^{k/2}$, $T(\mathbf{x}_i) = \mathbf{x}_i$ and $\alpha(\eta) = \eta^T\eta$.

By allowing the input data to be from any exponential family distribution [13]:

$$P(\mathbf{x}_i|\eta) = h(\mathbf{x}_i)\exp(\eta^T T(\mathbf{x}_i) - \alpha(\eta)), \quad \eta = \sum_j \mathbf{d}_j[j]a_{ij}, \quad (14)$$

where we use the basis vectors $\mathbf{d}_j[j]$ and activations a_{ij} to construct the natural parameter, following Lee et al. [13], we can get exponential family sparse coding.

Given the unlabeled data $\{\mathbf{x}_i^u\}_{i=1}^n$ in the auxiliary domain, following Eq. (12), the dictionary and activations are learned by solving the following optimization problem [13]:

$$\begin{aligned} \min_{\mathbf{D},\{\mathbf{a}_j\}} \quad & \sum_i -\log h(\mathbf{x}_i) - \mathbf{a}_i^T\mathbf{D}^T T(\mathbf{x}_i) + \alpha(\mathbf{D}\mathbf{a}_i) + \lambda\sum_{ij} a_{ij} \\ s.t. \quad & \|\mathbf{d}_j\| \le 1, \ \forall\ 1 \le j \le r \end{aligned} \quad (15)$$

Because the exponential family distributions guarantee the convexity of $-\log P(\mathbf{x}_i|\eta)$ with respect to η, it can be verified that the optimization objective in Eq. (15) is convex with respect to $\mathbf{D}$ for fixed $\{\mathbf{a}_i\}$, and with respect to $\mathbf{a}_i$ for fixed $\mathbf{D}$. Thus, we can use an alternative minimization method to solve Eq. (15). In this work, we use the algorithm introduced by Lee et al. [13].

Once the dictionary is learned by solving Eq. (15), new representations for the data vectors in the target domain can be learned by solving Eq. (5), from which classification is conducted as in Sect. 2.4.

4 Experimental results

In this section, we evaluate the proposed automatic TR record categorization system via self-taught learning by classifying the real depression homework data. We implement the self-taught learning model by using both traditional Gaussian sparse coding and exponential family sparse coding.

Data description The depression data contains 36 homework texts. Upon the content, these homeworks are manually divided into the four major thinking error categories as described in Table 2.

Table 2 Data distribution

Category	Number of homework texts
MISPERCEPTIONS	2
MAKING GUESSES	20
TUNNEL VISION	3
ABSOLUTES	11

In addition, we also use a large collection of free text as auxiliary data. In our experiments, we use the Yahoo data set, which was described in [23] coming from the "yahoo.com" domain. We use the "science" topic, which contains 6345 web pages.

4.1 Improved data representation via self-taught learning for TR categorization

We first evaluated the effectiveness of the learned data representation by self-taught learning in TR categorization. We compared the results of the self-taught learning method against the results of the traditional SVM approach using the original data representation directly abstracted from raw texts, i.e., using $\{\mathbf{x}_i^l\}_{i=1}^n$. A large amount of text data in the auxiliary domain was not used by our SVM. We implemented the self-taught learning method using traditional Gaussian sparse coding as well as exponential family sparse coding by assuming that the data came from a Poisson distribution following the approach by Lee et al. [13]. For exponential family sparse coding, instead of tf-idf vectors, we used the word count method to construct the data vector, in which each entry is a count of the number of appearances of the corresponding keyword. This yields a discrete representation of each TR, which is definitely simpler and less expensive for data pre-processing.

For our SVM implementation, we chose three types of kernels, the linear, polynomial, and Gaussian kernels. For the polynomial kernel, we fine-tuned the polynomial order of the polynomial kernel by searching the grid of $\{1, 2, \ldots, 10\}$; for the Gaussian kernel, i.e., $\mathcal{K}(\mathbf{x}_i, \mathbf{x}_j) = e^{-\gamma\|\mathbf{x}_i - \mathbf{x}_j\|^2}$, we fine-tuned the parameter γ by searching the grid of $\{10^{-5}, 10^{-4}, \ldots, 1, \ldots, 10^4, 10^5\}$. We set the tradeoff C in Eq. (6) to be 1. In addition, we report the classification results by the (k-NN) method as a baseline.

Experimental results We performed standard twofold cross-validation to evaluate the classification performance of the proposed method. Each experiment was repeated 50 times. The average classification accuracies of the compared methods are reported in Fig. 3.

A first glance at the results shows that the classifications of the enriched semantic representation by various self-taught learning methods outperform those on the original data representation. This observation is consistent with our theoretical analysis in that our new method exploits the information from the texts in the auxiliary domain, which are beneficial to the classification of the texts in the target domain. Our results suggest that the proposed automatic patient TR categorization system is not only cost-effective but may also be useful for diagnosis of patients and training of new therapists.

A more careful analysis on the results suggests that the data representations obtained from exponential family (Poisson distribution) sparse coding have better classification results than from traditional Gaussian sparse coding. This clearly demonstrates the usefulness of the non-Gaussian distribution assumption about the input data made based on the nature of text data.

4.2 Improved TR categorization via self-taught learning

Because the main goal of self-taught learning is to exploit the large amount auxiliary data when target data are very expensive to obtain, in this subsection we evaluate its effectiveness to make use of additional text data which can be obtained with very low, or even no, cost. We compare it to two conventional machine learning schemes including semi-supervised learning and transfer learning. For the former, we implement the Transductive SVM (TSVM) [10] method; for the latter, we implement the dyadic knowledge transfer (DKT) [26] method. Again, we evaluate the compared methods by twofold cross-validation, for which we report the average classification accuracies in Table 3.

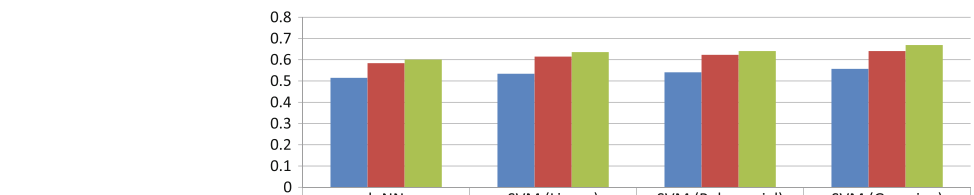
First, from Table 3 we can see that self-taught learning method and DKT method clearly outperform the TSVM method. This observation is consistent with the theoretical

Table 3 Comparison of self-taught learning against semi-supervised learning (TSVM) and transfer learning (DKT) for TR categorization

Method	Accuracy
TSVM	0.541
DKT	0.621
Self-taught learning	**0.669**

Bold value indicates the best result of all methods

Fig. 3 Classification accuracies by the compared methods



	k-NN	SVM (Linear)	SVM (Polynomial)	SVM (Gaussian)
Raw Data	0.515	0.534	0.541	0.557
Self-taught learning	0.584	0.615	0.623	0.641
Exponential family self-taught learning	0.601	0.636	0.641	0.669

formulations of these methods in that the former two methods are transfer learning methods while the latter one is a semi-supervised learning method. Because semi-supervised learning method requires the additional unlabeled data to be from the same distribution as the labeled one, which, however, cannot be generally satisfied in a lot of real-world applications, it is not a suitable learning scheme for cost-effective TR categorization. Second, we also notice that self-taught learning is better than DKT method. This is because DKT method transfers knowledge across different domains via nonnegative matrix factorizations, which implicitly assume that the input data are drawn from a Gaussian distribution. In contrast, exponential family (poisson distributed) self-taught learning method relaxes the assumption and is able to better simulate the text data, which thereby can achieve improved classification results. In summary, self-taught learning with exponential family sparse coding is an effective method to simultaneously improve the TR categorization performance and reduce the training cost.

5 Conclusions

In this paper, we presented a general framework using self-taught learning to represent the TR text in a new domain with respect to a content rich and discriminative semantic prototype sets learned from a huge amount of text obtained from cheap resources, such as those from Internet. The proposed method addressed the lack of the training data, including the data themselves and their associated labels. Through incorporating the useful information contained in the auxiliary domains, the classification performance in the target domain, i.e., the TR records we wish to categorize, is improved. Because the training of the proposed model does not require a large amount of labeled data in the domain of interest to achieve decent classification accuracy, our new system is cost effective. By further considering the discrete nature input data, instead of using traditional Gaussian sparse coding in self-taught learning, we use the exponential family sparse coding to better simulate the distribution of input data. Promising experimental results in the empirical studies demonstrate the effectiveness of the proposed method.

References

1. Arbib M (2003) The handbook of brain theory and neural networks. The MIT Press, Cambridge
2. Bishop C, service SO (2006) Pattern recognition and machine learning, vol 4. Springer, New York
3. Cristianini N, Shawe-Taylor J (2000) An introduction to support vector machines: and other kernel-based learning methods. Cambridge University Press, Cambridge
4. Dai W, Yang Q, Xue G, Yu Y (2008) Self-taught clustering. In: ICML
5. Duan K, Keerthi S (2005) Which is the best multiclass SVM method? An empirical study. Multiple Classif Syst 3541:278–285
6. Goodman J (2004) Exponential priors for maximum entropy models. In: Proceedings of the HLT-NAACL, pp 305–312
7. Hand D, Yu K (2001) Idiot's BayesNot So Stupid After All? Int Stat Rev 69(3):385–398
8. Hilbe J (2009) Logistic regression models. CRC Press, New York
9. Hsu C, Lin C (2002) A comparison of methods for multiclass support vector machines. IEEE Trans Neural Netw 13(2): 415–425
10. Joachims T (1999) Transductive inference for text classification using support vector machines. In: International conference on machine learning, pp 200–209
11. Kressel U (1999) Pairwise classification and support vector machines. Advances in kernel methods: support vector learning, pp 255–268
12. Lee H, Battle A, Raina R, Ng A (2007) Efficient sparse coding algorithms. In: NIPS
13. Lee H, Raina R, Teichman A, Ng A (2009) Exponential family sparse coding with applications to self-taught learning. IJCAI09, pp 1113–1119
14. Liu J, Ji S, Ye J (2009) Multi-task feature learning via efficient $\ell_{2,1}$-norm minimization. In: Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence, AUAI Press, Arlington, pp 339–348
15. Mairal J, Bach F, Ponce J, Sapiro G, Zisserman A (2009) Supervised dictionary learning. In: NIPS, pp 1033–1040
16. Manning C, Raghavan P, Schütze H, Corporation E (2008) Introduction to information retrieval, vol 1. Cambridge University Press, Cambridge
17. Pan S, Yang Q (2009) A survey on transfer learning. IEEE TKDE
18. Raina R (2009) Self-taught learning. PhD thesis of Stanford University
19. Raina R, Battle A, Lee H, Packer B, Ng A (2007) Self-taught learning: transfer learning from unlabeled data. In: ICML
20. Seeger M (2008) Bayesian inference and optimal design for the sparse linear model. The Journal of Machine Learning Research 9:759–813
21. Sumathi S, Sivanandam S (2006) Introduction to data mining and its applications. Springer, New York
22. Tibshirani R (1996) Regression shrinkage and selection via the lasso. J R Stat Soc B 58(1):267–288
23. Ueda N, Saito K (2002) Single-shot detection of multiple categories of text using parametric mixture models. In: Proceedings of SIGKDD, pp 626–631
24. Wang H, Huang H, Basco M, Lopez M, Makedon F (2011) Cost effective depression patient thought record categorization via self-taught learning. In: Proceedings of the 4th international conference on pervasive technologies related to assistive environments (PETRA 2011), p 41. ACM, New York
25. Wang H, Huang H, Nie F, Ding C (2011) Cross-language web page classification via dual knowledge transfer using nonnegative matrix tri-factorization. In: Proceedings of the 34th international ACM SIGIR conference on research and development in Information, pp 933–942. ACM, New York
26. Wang H, Nie F, Huang H, Ding C (2011) Dyadic transfer learning for cross-domain image classification. In: IEEE international conference on computer vision (ICCV), pp 551–556
27. Zhu X (2006) Semi-supervised learning literature survey. Technical report, University of Wisconsin-Madison