Emotion Distribution Learning from Texts

Deyu Zhou, Xuan Zhang, Yin Zhou, Quan Zhao, Xin Geng*
MOE Key Laboratory of Computer Network and Information Integration
School of Computer Science and Engineering
Southeast University, Nanjing, China
{d.zhou, zhouying1, xuanzhang, zhaoquan, xgeng}@seu.edu.cn

Abstract

The advent of social media and its prosperity enable users to share their opinions and views. Understanding users' emotional states might provide the potential to create new business opportunities. Automatically identifying users' emotional states from their texts and classifying emotions into finite categories such as *joy*, *anger*, *disgust*, etc., can be considered as a text classification problem. However, it introduces a challenging learning scenario where multiple emotions with different intensities are often found in a single sentence. Moreover, some emotions co-occur more often while other emotions rarely coexist. In this paper, we propose a novel approach based on emotion distribution learning in order to address the aforementioned issues. The key idea is to learn a mapping function from sentences to their emotion distributions describing multiple emotions and their respective intensities. Moreover, the relations of emotions are captured based on the Plutchik's wheel of emotions and are subsequently incorporated into the learning algorithm in order to improve the accuracy of emotion detection. Experimental results show that the proposed approach can effectively deal with the emotion distribution detection problem and perform remarkably better than both the state-of-the-art emotion detection method and multi-label learning methods.

1 Introduction

The advent of social media and its prosperity enable the creation of massive online user-generated content including opinions and product reviews. Analyzing such user-generated content allows the detection of users' emotional states, which might be potentially useful for downstream applications such as brand watching, product recommendation, and detection of health-related issues, etc. Based on the way emotions are represented, computational models for emotion analysis can be categorized into dimensional models and categorical models (Calvo and D'Mello, 2010). Dimensional approaches (Russell, 2003) emphasize the fundamental dimensions of valence and arousal in understanding emotional experience, which have long been studied by emotion theorists. Categorical models (Gupta et al., 2013) involve the use of a categorical representation, in which emotions are represented by a number of labels. For example, Ekman's basic emotion set (Ekman, 1992) consists of anger, disgust, fear, happiness, sadness and surprise. An example of a sentence and the annotated emotions can be found in Table 1.

Sentence	Trains crash near Thai resort town					
Emotions	anger	disgust	fear	joy	sadness	surprise
	2	0	62	0	90	10

Table 1: An example of a sentence containing emotions selected from SemEval 2007 Task#14, Affective Text, where each of the six emotions are indicated using a score of [0, 100].

Considering each basic emotion as class label for the sentence, emotion detection can be treated as a classification problem. There is a large body of prior work on emotion classification (Mishne and de Rijke, 2006; Lin and He, 2009; Quan et al., 2015; Wang and Pal, 2015). By choosing the strongest emotion as the emotion label for the sentence, most of

*Corresponding author

Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 638–647,
Austin, Texas, November 1-5, 2016.

classification approaches are based on single-label learning. However, as shown in Table 1, a sentence might contain multiple emotions with varying intensities. Although, some lexicon-based approach such as (Wang and Pal, 2015) can output multiple emotions with intensities using non-negative matrix factorization. It can only guarantee convergence to a local minimum, which is prohibitive on the large, realistically-sized emotion detection problem.

Machine learning methods such as multi-label learning (MLL) can be employed to identify multiple emotions for each sentence (Zhang and Zhou, 2014). MLL usually selects a threshold, then labels emotions with scores higher than the threshold as relevant and the others as irrelevant. However, these methods are not able to learn the intensity of each emotion. To address this problem, a new machine learning paradigm called Label Distribution Learning (LDL) (Geng, 2016) was proposed in recently years. Similarly, in this paper, we propose an emotion distribution learning (EDL) algorithm. Different from the previous approaches, EDL assumes that each sentence contains a mixture of basic emotions with different intensities. Using categorical model, we can label each sentence with an emotion vector where each element corresponds to one basic emotion and the value of each element indicates the intensity of the emotion. We require that each vector element has a value between 0 and 1 and they sum up to 1. By doing so, the emotion vectors can be considered as emotion distributions and the proposed EDL algorithm aims to learn the mapping from sentences to their corresponding emotion distributions by minimizing the differences between the true distributions and the predicted distributions. Both the single-label learning and MLL can be considered as special cases of EDL in emotion detection. Moreover, as some emotions co-occur more often while others rarely co-exist, the relations between basic emotions are captured according to the Plutchik's wheel of emotions theory (Plutchik, 1980) and are incorporated in the learning framework as constraints in order to improve the accuracy of emotion detection.

Our work makes the following contributions:

- We propose a novel approach based on emotion distribution learning to identify multiple emotions with their intensities from texts. To the best of our knowledge, it is the first attempt to identify both emotions and intensities in the distribution learning framework.
- The relations between basic emotions are incorporated into the learning framework as constraints to improve the emotion detection accuracy. To avoid the incorporation of noisy information from the training data, the relation constraint is set based on the Plutchik's wheel of emotions theory.
- Experimental results show that the proposed approach can effectively deal with the emotion distribution detection problem and perform remarkably better than the state-of-the-art multi-label learning methods and emotion detection method.

2 Related Work

In general, emotion classification can be approached by two types of methods, lexicon-based or corpus-based. Lexicon-based approaches rely on emotion lexicons consisting of words and their corresponding emotion labels for detecting emotions from text. For example, WordNetAffect (Strapparava and Valitutti, 2004) was constructed by extending Wordnet, a lexical database of English terms, with information on affective terms. EmoSenticNet assigns six WordNetAffect emotion labels to SenticNet concepts (Poria et al., 2013), which can be thought of as an expansion of WordNetAffect emotion labels to a larger vocabulary. Many approaches were proposed based on emotion lexicons. For example, (Aman and Szpakowicz, 2007) classified emotional and non-emotional sentences using the constructed emotion lexicon. (Choudhury et al., 2012) employed a classifier to detect human affective states in social media. (Wang and Pal, 2015) proposed a model with several constraints based on an emotion lexicon for emotion classification.

Corpus-based methods aim to train supervised classifiers from annotated training data where each sentence or document is labelled with an emotion class. (Mishne and de Rijke, 2006) constructed models to predict the levels of various moods according to the language used by bloggers at a giv-

en time. (Aman and Szpakowicz, 2007) described an emotion annotation task of identifying emotion category, emotion intensity and the words/phrases that indicate emotions in text. Emotion classification was conducted using trained support vector machines. (Agrawal and An, 2012) proposed an unsupervised context-based approach to detect emotions from text at the sentence level. They computed an emotion vector for each potential affect bearing word based on the semantic relatedness between words and various emotion concepts. The scores are then tuned using the syntactic dependencies within the sentence structure. (Bao et al., 2009) proposed an emotion topic model by augmenting latent Dirichlet allocation with an intermediate emotion layer. (Quan et al., 2015) proposed a logistic regression model for social emotion detection. Intermediate hidden variables were also introduced to model the latent structure of input text corpora.

Our work is partly inspired by (Quan et al., 2015). However, our proposed approach differs from (Quan et al., 2015) in two aspects: 1) by introducing the emotion distribution learning framework, many different criteria can be used to measure the distance between the true distribution and the predicted distribution, such as squared $\mathcal{X}^2$, Euclidean, Jeffery's divergence apart from Kullback-Leibler divergence employed in logistic regression model. 2) the relations between basic emotions are captured based on the Plutchik's wheel of emotions theory to avoid the incorporation of any noisy information from the training data.

3 Emotion Distribution Learning

3.1 Problem Setting

As have discussed in section 1, one sentence might contain one or more emotions, and each emotion has its own intensity. We use d_x^y to indicate the intensity of emotion y for sentence x, where $x \in \mathcal{X}$ and $y \in \mathcal{Y}$. The emotion intensity is normalized to make $d_x^y \in [0,1]$ and $\sum_y d_x^y = 1$ to constitute the emotion distribution.

Note that d_x^y denotes the proportion that y accounts for in a full emotion distribution of x. It is different from the probability of y being a correct emotion label for x. Probability distribution implies that only one emotion label is correct for each sentence, while emotion distribution allows multiple emotions in one sentence. The goal of EDL is to learn a mapping from sentences $\mathcal{X} = \mathbb{R}^m$ to the distributions over a finite set of labels $\mathcal{Y} = \{y_1, y_2, ...y_c\}$. Each label represents one of the basic emotions.

3.2 Learning

Given a training set $P = \{(x_1, E_1), (x_2, E_2), ..., (x_n, E_n)\}$, where $x_i \in \mathcal{X}$ is a sentence and $E_i = \{d_{x_i}^{y_1}, d_{x_i}^{y_2}, ..., d_{x_i}^{y_c}\}$ is the emotion distribution associated with x_i. The goal of EDL is to learn a conditional probability mass function $p(y|x)$ from P, where $x \in \mathcal{X}$ and $y \in \mathcal{Y}$. Assuming that $p(y|x)$ is a parametric model $p(y|x;\theta)$, where θ are model parameters, many different criteria can be used to measure the distance between two distributions, such as Squared $\mathcal{X}^2$, Euclidean, Jeffery's divergence, Kullback-Leibler (K-L) divergence and so on. Here we use Divergence defined by

$$D_J(Q_a||Q_b) = 2\sum_j \frac{(Q_a^j - Q_b^j)^2}{(Q_a^j + Q_b^j)^2}$$

, where Q_a^j and Q_b^j are the j-th element of the two distributions Q_a and Q_b, respectively. Divergence is balanced, which makes $D_J(Q_a||Q_b)$ equal to $D_J(Q_b||Q_a)$. The formula above calculates the sum of all the distances between emotion intensities in the same position.

Then the optimal model parameters θ^* is determined by

$$\begin{aligned}
\theta^* &= \arg\min_\theta \left\{ \sum_i D_J(E_i||\hat{E}_i) + \frac{\xi_1}{n}\sum_{k,r}|\theta_{k,r}|_1 \right. \\
&\left. + \frac{\xi_2}{n}\sum_u\sum_{j,k}\omega_{jk}\|\theta_{u,j} - \theta_{u,k}\|_2^2 \right\} \\
&= \arg\min_\theta \left\{ 2\sum_{i,j}\frac{(d_{x_i}^{y_j} - p(y_j|x_i,\theta))^2}{(d_{x_i}^{y_j} + p(y_j|x_i,\theta))^2} \right. \\
&+ \frac{\xi_1}{n}\sum_{k,r}|\theta_{k,r}|_1 \\
&\left. + \frac{\xi_2}{n}\sum_u\sum_{j,k}\omega_{jk}\|\theta_{u,j} - \theta_{u,k}\|_2^2 \right\}
\end{aligned} \tag{1}$$

, where E_i is the ground truth emotion distribution of the i-th sentence and the $\hat{E}_i$ is the predicted one by $p(y|x_i;\theta)$. The second term is a regularizer to make the predicted emotion distribution sparse, and the third term considers the relationship between different emotions. As mentioned in section 1, some emotions often co-occur such as joy and love, and some rarely co-exist such as joy and anger. Therefore, the third term is employed to incorporate such prior knowledge. The weight ω_{jk} models the relationship between the *j*-th emotion and the *k*-th emotion in the distribution. In this paper, we capture the relationships between different emotions based on Plutchik's wheel of emotions (Plutchik, 1980) which is produced in psychology view. Plutchik's wheel of emotions includes several typical emotions and its eight sectors indicate eight primary emotion dimensions arranged as four pairs of opposites. We re-produce a wheel of eight emotions' relationships according to Plutchik's theory, which is shown in Figure 1.

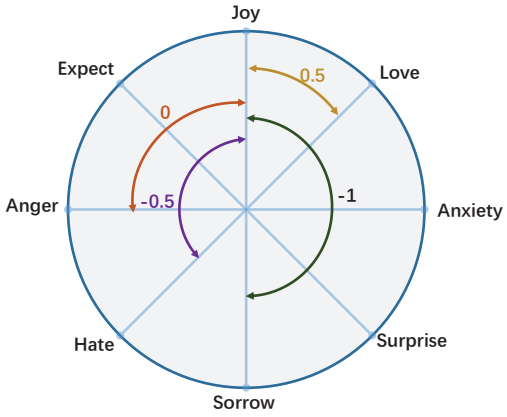


Figure 1: Plutchik's wheel of emotions.

In the emotion wheel, emotions sat at opposite end have an opposite relationship, while emotions next to each other are more closely related. We quantify the relations between each pair of emotions based on the angle between them in wheel of emotions (Plutchik, 2001). For example, emotion pairs with 180 degrees are opposite to each other, which are described by -1, while emotion pairs with 90 degrees are described by 0, meaning no relationship between them. Emotion pairs with 45 degrees have the relationship value of 0.5, while emotion pairs with 135 degrees have the relationship value of -0.5. Figure 2 shows the gray-scale image of the pair-wise relationships of emotions presented in Figure 1. In each cell, the darker the color is, the more similar the two emotions are.

As for $p(y|x;\theta)$, similar to (Geng, 2016), we assume it takes a maximum entropy model, i.e.,

$$p(y_k|x_i;\theta) = \frac{1}{\mathcal{Z}_i}\exp(\sum_r \theta_{kr}x_i^r) \tag{2}$$

, where $\mathcal{Z}_i = \sum_k \exp(\sum_r \theta_{kr}x_i^r)$ is the normalization factor, x_i^r is the *r*-th feature of x_i, and θ_{kr} is an element in θ. Substituting Equation 2 into Equation 1 yields the target function,

$$\begin{aligned} T(\theta) = 2\sum_{i,j}\left(1 - \frac{4\mathcal{Z}_i d_{x_i}^{y_j}\exp\left(\sum_r \theta_{jr}x_i^r\right)}{\left(\mathcal{Z}_i d_{x_i}^{y_j} + \exp\left(\sum_r \theta_{jr}x_i^r\right)\right)^2}\right) \\ + \frac{\xi_1}{n}\sum_{k,r}|\theta_{k,r}|_1 \\ + \frac{\xi_2}{n}\sum_u\sum_{j,k}\omega_{jk}\|\theta_{u,j} - \theta_{u,k}\|_2^2. \end{aligned} \tag{3}$$

The minimization of the function $T(\theta)$ can be effectively solved by the limited-memory quasi-Newton method (L-BFGS). The basic idea of L-BFGS is to avoid explicit calculation of the inverse Hessian matrix used in the Newton method. L-BFGS approximates the inverse Hessian matrix with an iteratively updated matrix instead of actually storing the full matrix. Here we follow the idea of an effective quasi-Newton method BFGS. Consider the second-order Taylor series of $T'(\theta) = -T(\theta)$ at the current estimate of the parameter vector $\theta^{(l)}$:

$$\begin{aligned} T'(\theta^{(l+1)}) \approx T'(\theta^{(l)}) + \nabla T'(\theta^{(l+1)})^T\Delta \\ + \frac{1}{2}\Delta^T H(\theta^{(l)})\Delta, \end{aligned} \tag{4}$$

where $\Delta = \theta^{(l+1)} - \theta^{(l)}$ is the update step, $\nabla T'(\theta^{(l)})$ and $h(\theta^{(l)})$ are the gradient and Hessian matrix of $T'(\theta^{(l)})$ at $\theta^{(l)}$, respectively. The minimizer of Equation 4 is

$$\Delta^l = -H^{-1}(\theta^{(l)})\nabla T'(\theta^{(l)}). \tag{5}$$

The line search Newton method uses $\Delta^{(l)}$ as the search direction $p^{(l)} = \Delta^{(l)}$ and updates model parameters by

$$\theta^{(l+1)} = \theta^{(l)} + \alpha^{(l)}p^{(l)}, \tag{6}$$

where the step length $\alpha^{(l)}$ is obtained from a line search procedure to satisfy the strong Wolfe conditions (Nocedal and Wright, 2006):

$$T'(\theta^{(l)}+\alpha^{(l)}p^{(l)}) \leq T'(\theta^{(l)})+c_1\alpha^{(l)}\nabla T'(\theta^{(l)})^T p^{(l)}$$
$$|\nabla T'(\theta^{(l)}+\alpha^{(l)}p^{(l)}| \leq c_2|\nabla T'(\theta^{(l)})^T p^{(l)}|,$$

where $0 < c_1 < c_2 < 1$. The idea of BFGS is to avoid explicit calculation of $H^{-1}(\theta^{(l)})$ by approximating it with an iteratively updated matrix B, i.e.

$$\begin{aligned} B^{(L+1)} &= (I-\rho^{(l)}s^{(l)}(u^{(l)})^T)\times B^{(l)} \\ &\quad \times(I-\rho^{(l)}u^{(l)}(s^{(l)})^T) \\ &\quad +\rho^{(l)}s^{(l)}(s^{(l)})^T \end{aligned}$$

where

$$\begin{aligned} s^{(l)} &= \theta^{(l+1)}-\theta^{(l)}, \\ u^{(l)} &= \nabla T'(\theta^{(l+1)})-\nabla T'(\theta^{(l)}), \\ \rho^{(l)} &= \frac{1}{s^{(l)}u^{(l)}}. \end{aligned}$$

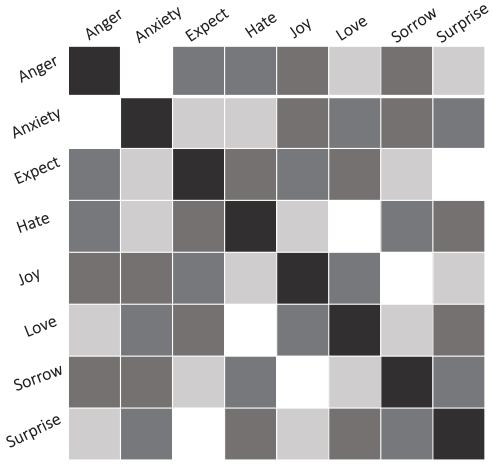


Figure 2: Gray-scale image of the pair-wise relationships of emotions shown in Figure 1.

As for the optimization of the target function $T(\theta)$, the computation of BFGS is mainly related to the first-order gradient of $T'(\theta)$, which can be achieved by

$$\begin{aligned} \frac{\partial T(\theta)}{\partial \theta_{jr}} &= \frac{4d_{x_i}^{y_j}p_{ij}(1-p_{ij})(d_{x_i}^{y_j}-p_{ij})}{(d_{x_i}^{y_j}+p_{ij})^3} \\ &\quad +\xi_1\sum_{k,r} sgn(\theta_{k,r}) \\ &\quad +\frac{1}{n}\xi_2\sum_k \omega_{jk}(\theta_j-2\theta_k), \end{aligned} \quad (7)$$

where $p_{ij} = \frac{1}{Z_i}\exp(\sum_r \theta_{jr}x_i^r)$. Thus it performs more efficiently than the standard line search Newton method.

In order to compare with the MLL methods, labels in the predicted distribution need to be divided into two sets, i.e, the relevant and irrelevant sets. For this purpose, an extra virtual label y_0 is added into the label set, i.e., the extended label set $\mathcal{Y}' = \mathcal{Y} \cup \{y_0\}$=$\{y_0, y_1, y_2 ... y_c\}$. Using the new extended label set in the training process, the optimal parameter vector θ^* is learned. As y_0 is the label that distinguishes the relevant and irrelevant emotions directly, it is initialized as the threshold used in MLL. Given a sentence x', its emotion distribution is predicted by $p(y|x';\theta^*)$. The intensity value of y_0 splits the predicted distribution into two sets. The emotions with the intensity value higher than y_0's are regarded as the relevant emotions, and the rest emotions are regarded as irrelevant ones. Therefore, EDL in fact implements the function of MLL without the need of setting the threshold manually.

4 Experiments

4.1 Setup

We evaluate the proposed approach on the RenCECps corpus (Quan and Ren, 2010). It contains $35,096$ sentences selected from blogs in Chinese. Each sentence is annotated with 8 basic emotions, such as *anger*, *anxiety*, *expect*, *hate*, *joy*, *love*, *sorrow* and *surprise*, together with their emotion scores. Higher score represents higher emotion intensity. We use $AS_i(j)$ to represent the score of emotion j in sentence i. Given a sentence x_i, the intensity of emotion j is calculated by $d_{x_i}^{y_j} = \frac{AS_i(j)}{\sum_k AS_i(k)}$. By doing so, each intensity value fulfills $d_{x_i}^{y_j} \in [0,1]$ and $\sum_y d_{x_i}^{y_j} = 1$.

For each sentence, features are extracted using recursive auto-encoders (RAEs) (Socher et al., 2011). RAEs are neural networks that represent meanings of fixed-size inputs in the reduced dimensional space. For example, each word in a sentence is represented using a vector $w \in \mathbb{R}^d$, and the RAE method reduces the entire sentence to a single vector of size $\mathbb{R}^d$. Sentences are sequences of words that can be represented by a binary tree structure. The words are the leaves of the tree and their combined grouping is used to get a notion of the meaning of the sentence.

The internal nodes of the tree correspond to the combined meaning of the nodes underneath them. Each internal node is also represented in the same manner as individual words in the form of a vector $\hat{w} \in \mathbb{R}^d$. These internal nodes are the hidden representations of the neural network. In the RAE model, the vocabulary is stored in an embedding matrix $V \in \mathbb{R}^d \times D$ where D is the cardinality of the vocabulary. Typically, each word $w \in V$ is initialized independently following a Gaussian distribution $w_i \sim N(0, \gamma^2)$. In our experiment, we set the dimension of each sentence representation to 100.

We build a gray-scale image shown in Figure 3 by computing the correlation coefficient of the emotions from the Ren-CECps corpus. It can be observed that Figure 3 is quite similar to Figure 2, which shows that our proposed way in capturing the relations between emotions is inline with what have been revealed by the emotion annotations in the Ren-CECps corpus.

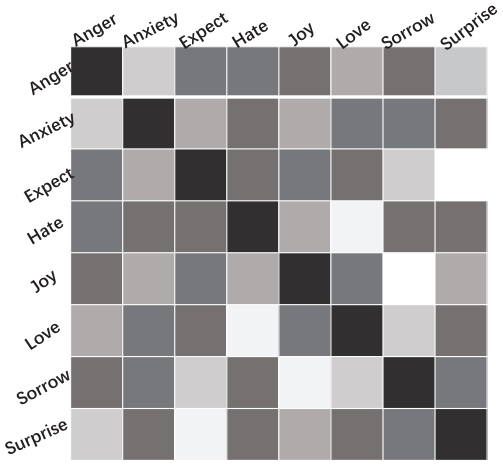


Figure 3: Gray-scale image of the pair-wise relations of the emotions in the Ren-CECps corpus.

4.2 Experimental Results

As the output of EDL is a distribution, a natural choice of criteria is the averaged similarity or distance between the actual emotion distribution and the predicted distribution. There are many metrics that can be applied to measure the distance between two distributions. In this paper six of them are used to evaluate the results of EDL, i.e, Euclidean, Sϕrensen, Squared $\mathcal{X}^2$, KL divergence, Intersection and Fidelity, as suggested in (Geng and Ji, 2013).

	Name	Formula
Distance	Euclidean	$Euclidean(P,Q) = \sqrt{\sum_{j=1}^{c}(P_j - Q_j)^2}$
	Sϕrensen	$S\phi rensen(P,Q) = \frac{\sum_{j=1}^{c}\lvert P_j - Q_j\rvert}{\sum_{j=1}^{c}(P_j + Q_j)}$
	Squared $\mathcal{X}^2$	$Squared\mathcal{X}^2(P,Q) = \sum_{j=1}^{c}\frac{(P_j - Q_j)^2}{P_j + Q_j}$
	Kullback-Leibler (KL)	$K\text{-}L(P,Q) = \sum_{j=1}^{c} P_j \ln \frac{P_j}{Q_j}$
Similarity	Intersection	$Intersection(P,Q) = \sum_{j=1}^{c}\min(P_j, Q_j)$
	Fidelity	$Fidelity(P,Q) = \sum_{j=1}^{c}\sqrt{P_j Q_j}$

Table 2: Evaluation criteria for the Label Distribution Learning (LDL) methods.

Name	Formula
Hamming Loss	$hloss(h) = \frac{1}{P}\sum_{i=1}^{P}\lvert h(x_i) \triangle Y_i\rvert$
One error	$one\text{-}error(f) = \frac{1}{P}\sum_{i=1}^{P}[\arg\max_{y \in Y} f(x_i, y)] \notin Y_i$
Coverage	$Coverage(f) = \frac{1}{P}\sum_{i=1}^{P}\max_{y \in Y_i} rank_f(x_i, y) - 1$
Ranking Loss	$rloss(f) = \frac{1}{P}\sum_{i=1}^{P}\frac{1}{\lvert Y_i\rvert\lvert\bar{Y}_i\rvert} \cdot \lvert R\rvert, Where$ $R = (y', y'')\lvert f(x_i, y') \leq f(x_i, y''), (y', y'') \in Y_i \times \bar{Y}_i$
Average Precision	$Average(f) = \frac{1}{P}\sum_{i=1}^{P}\frac{1}{\lvert Y_i\rvert}\sum_{y \in Y_i}\frac{\lvert P_i\rvert}{rank_f(x_i, y)}, where$ $P_i = y'\lvert rank_f(x_i, y') \leq rank_f(x_i, y), (y)' \in Y_i$

Table 3: Evaluation criteria for the MLL methods.

The formulae of the six criteria are summarized in Table 4.2. Note that the virtual label y_0 is removed before evaluation.

As EDL can output both the relevant emotions and their respective emotion intensities, MLL can be seen as a special case of EDL that it only outputs emotion labels but not their intensities. Several evaluation criteria typically used in MLL can also be used to measure EDL's ability of distinguishing relevant emotions from irrelevant ones, including hamming loss, one error, coverage, ranking loss, and average precision as suggested by (Zhang and Zhou, 2014), which are summarized in Table 4.2. Hamming loss evaluates how many times an emotion label is misclassified. One-error evaluates the fraction of sentences whose top-ranked emotion is not in the relevant emotion set. Coverage evaluates how many steps are needed to move down the ranked emotion list so as to cover all the relevant emotions of the example. Ranking loss evaluates the fraction of reversely ordered emotion pairs. Average precision evaluates the average fraction of the relevant emotions ranked higher than a particular emotion $y \in \mathcal{Y}$.

For each algorithm, ten-fold cross validation is conducted. EDL is first compared with four existing Label Distribution Learning (LDL) methods (Geng,

Algorithm	Evaluation Criterion					
	Euclidean(↓)	Sørensen(↓)	Squared $\mathcal{X}^2$(↓)	K-L(↓)	Intersection(↑)	Fidelity(↑)
EDL	**0.2361±0.0057**	**0.2346±0.0061**	**0.1780±0.0037**	**0.2067±0.0046**	**0.7654±0.0046**	**0.9523±0.0019**
AA-KNN (Geng, 2016)	0.2948±0.0101•	0.2941±0.0123•	0.2688±0.0102•	0.3163±0.0087•	0.7059±0.0078•	0.9258±0.0090•
PT-Bayes (Geng, 2016)	0.3295±0.0125•	0.3288±0.0158•	0.2826±0.0115•	0.3263±0.0238•	0.6711±0.0241•	0.9238±0.0060•
PT-SVM (Geng, 2016)	0.3614±0.0869•	0.3625±0.0145•	0.3415±0.0089•	0.4073±0.0209•	0.6375±0.0099•	0.9069±0.0073•
AA-BP (Geng, 2016)	0.3299±0.0159•	0.3430±0.0264•	0.2885±0.0251•	0.3406±0.0092•	0.6569±0.0166•	0.9229±0.0056•
emoDetect (Wang and Pal, 2015)	0.3333±0.0678•	0.3468±0.0719•	0.2928±0.0674•	0.3463±0.0790•	0.6532±0.0719•	0.9212±0.0180•

Table 4: Experimental results in comparison with the LDL methods and the emotion detection approach.

Algorithm	Evaluation Criterion				
	Average Precision(↑)	Coverage(↓)	Hamming Loss(↓)	One Error(↓)	Ranking Loss(↓)
EDL	**0.6419±0.0235**	**2.1412±0.0235**	**0.1772±0.0568**	**0.5239±0.0945**	**0.2513±0.0560**
ML-KNN (Zhang and Zhou, 2014)	0.5917±0.0742•	2.448±0.0981•	0.2459±0.0781•	0.5339±0.0954•	0.2908±0.0431•
LIFT (Zhang, 2011)	0.5979±0.0891•	2.4267±0.0492•	0.1779±0.0597•	0.5131±0.0666•	0.2854±0.0427•
Rank-SVM (Zhang and Zhou, 2014)	0.5738±0.0892•	2.5861±0.0777•	0.2485±0.0458•	0.5603±0.0921•	0.3055±0.0579•
MLLOC (Huang and Zhou, 2012)	0.4135±0.0568•	3.6994±0.0764•	0.1850±0.0659•	0.6971±0.0924•	0.4742±0.0734•
BP-MLL (Zhang and Zhou, 2006)	0.4791±0.0999•	3.3773±0.0681•	0.2108±0.0986•	0.6316±0.0988•	0.4293±0.0956•
ECC (Read et al., 2011)	0.5121±0.0892•	2.7767±0.0876•	0.1812±0.0945•	0.6969±0.0598•	0.3281±0.0659•

Table 5: Experimental results in comparison with the MLL methods.

2016), i.e., PT-Bayes, PT-SVM, AA-KNN, AA-BP. k in AA-KNN is set to 8. Linear kernel is used in PT-SVM. The number of hidden-layer neurons for AA-BP is set to 60. The evaluation results of our proposed approach in comparison to the LDL baselines are presented in Table 4.2. For all the measures, "↓" indicates "the smaller the better", while "↑" indicates "the larger the better". The best performance on each measure is highlighted by boldface. The two-tailed t-tests with 5% significance level are performed to see whether the differences between EDL and the baselines are statistically significant. We use • to indicate significance difference. As the state-of-the-art emotion detection method proposed in (Wang and Pal, 2015) can output the emotion distributions based on a dimensional reduction method, we present its experimental results on the Ren-CECps corpus in the last row of Table 4.2. It can be observed that EDL performs significantly better than all the baseline LDL methods and the state-of-the-art emotion detection approach on all criteria considered here.

Since EDL can be seen as an extension of MLL, EDL is compared with 7 widely used MLL methods using the virtual label y_0, namely ML-KNN (Zhang and Zhou, 2014), ECC (Read et al., 2011), MLLOC (Huang and Zhou, 2012), LIFT (Zhang, 2011), ML-RBF (Zhang, 2009), Rank-SVM (Zhang and Zhou, 2014), BP-MLL (Zhang and Zhou, 2006). Among the compared algorithms, ML-kNN is derived from the traditional k-nearest neighbor (kNN) algorithm. Maximum a posteriori (MAP) principle is used to determine which emotion set is related to the given sentence. CC (classifier chains method) overcomes the limitations of BR and performs better but requires more computations. ECC (ensemble classifier chains) applies classifier chains in an ensemble framework and obtains high predictive performances. MLLOC (Multi-label LOcal Correlation) tries to exploit emotion correlations in the expression data locally. The global discrimination fitting and local correlation sensitivity are incorporated into a unified framework, and solution for the optimization are developed. Rank-SVM provides a way of controlling the complexity of the overall learning system while having a small empirical error. The architectures of Rank-SVM is based on linear models of Support Vector Machines (SVM) (Boser et al., 1992). LIFT constructs features specific to each emotion by conducting clustering analysis on its positive or negative instances, and then performs training and testing by querying the clustering results (Zhang, 2011). BP-MLL is derived from the famous backpropagation algorithm through employing a novel error function capturing the characteristics of multi-label learning, i.e., the emotions belonging to a sentence should be ranked higher than those not belonging to that sentence (Zhang and

Zhou, 2006).

The virtual label y_0 used in EDL and the threshold value used in MLL are all set to 2.5. Besides, the ε, ξ_1 and ξ_2 are set as 0.25, 0.0001, 0.1 respectively. For the MLL methods, the value of k is set to 8 in ML-KNN, ratio is 0.02 and μ is 2 in ML-RBF. Linear kernel is used in LIFT. Rank-SVM uses the RBF kernel with the width σ equals to 1. The evaluation results of the proposed approach in comparison to all MLL baselines are presented in Table 4.2. EDL performs best on all evaluation measures. It verifies the advantage of EDL owing to the consideration of varying intensity of the basic emotions.

4.3 Further Analysis

To fully understand the emotion detection results, we use word cloud (Harris, 2011) to output the top 30 frequent words in the testing data for the emotion *love* and *anxiety* based on the annotation as shown in the left part of Figure 4. We also output the top 30 frequent words for the two emotions based on the prediction generated by EDL as shown in Figure 4 's right part. It can be observed that most words based on prediction indeed express their associated emotions. For example, word "like" delivers the emotion of *love* (right part of Figure 4(a)) and word "problem" tells *anxiety* (right part of Figure 4(b)). Moreover, the annotation and the prediction share 20 out of the top 30 most frequent words for the emotion *love* such as "friend", "joy", "happiness", etc as shown in the middle of Figure 4(a) and 19 out of 30 for the emotion of *anxiety* (the middle of Figure 4(b). It demonstrates that EDL can learn emotions from text precisely.

To investigate the emotion distributions generated by EDL, a sentence from the Ren-CECps corpus together with the emotion distribution output by EDL is illustrated in Figure 5. The ground truth emotion distribution is obtained by normalizing the scores and the virtual label y_0. As can be seen, the curve of the predicted emotion distribution is very similar as the ground truth distribution, which demonstrates that EDL can learn the varying intensities of all the basic emotions well.

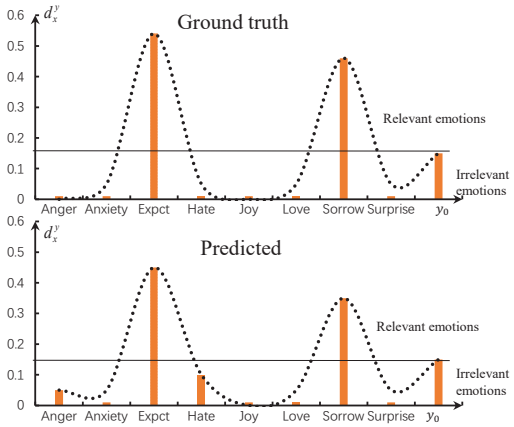


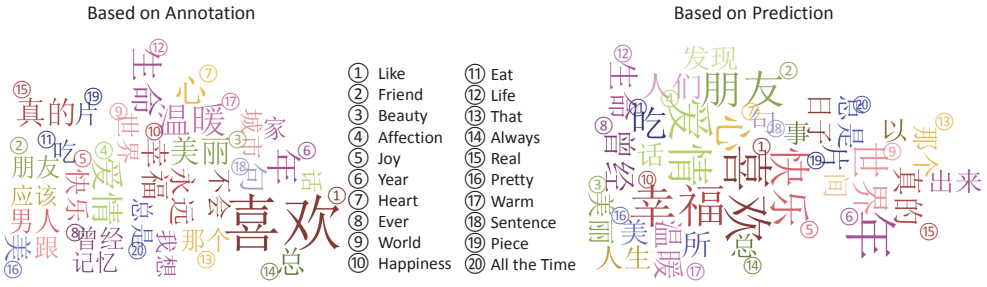
Figure 5: A sentence with the emotion distribution predicted by EDL.

5 Conclusions and Future Work

In this paper, we have proposed a novel approach based on EDL to identify multiple emotions with their intensities from texts. Moreover, the relations between basic emotions is incorporated in the learning framework as constraints to improve the learning accuracy. Experimental results show that the proposed approach can effectively deal with the emotion distribution detection problem and perform remarkably better than the state-of-the-art multi-label learning methods and the emotion detection method. In future work, we will investigate the efficiency of the proposed approach in other datasets and explore other methods in capturing the inter-relations of emotions.

Acknowledgments

This work was funded by the National Natural Science Foundation of China (61273300, 61232007, 61528302, 61622203), the Jiangsu Natural Science Funds for Distinguished Young Scholar (BK20140022), the Natural Science Foundation of Jiangsu Province of China (BK20161430), and the Collaborative Innovation Center of Wireless Communications Technology.



(a) Love

(b) Anxiety

Figure 4: Top 30 frequent words for the emotion *love* and *anxiety* based on the annotation or the prediction.

References

Ameeta Agrawal and Aijun An. 2012. Unsupervised emotion detection from text using semantic and syntactic relations. In *Proceedings of the 2012 IEEE/WIC/ACM International Joint Conferences on Web Intelligence and Intelligent Agent Technology*, pages 346–353.

Saima Aman and Stan Szpakowicz. 2007. Identifying expressions of emotion in text. *Lecture Notes in Computer Science*, 4629:196–205.

S. Bao, Shengliang Xu, Li Zhang, Rong Yan, Zhong Su, Dingyi Han, and Yong Yu. 2009. Joint emotion-topic modeling for social affective text mining. In *Proceedings of the Ninth IEEE International Conference on Data Mining*, pages 699–704.

Bernhard E. Boser, Isabelle M. Guyon, and Vladimir N. Vapnik. 1992. A training algorithm for optimal margin classifiers. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, pages 144–152.

R.A. Calvo and S. D'Mello. 2010. Affect detection: An interdisciplinary review of models, methods, and their applications. *Affective Computing, IEEE Transactions on*, 1(1):18–37.

Munmun De Choudhury, Michael Gamon, and Scott Counts. 2012. Happy, nervous or surprised? classification of human affective states in social media. In *Proceedings of the Sixth International AAAI Conference on Weblogs and Social Media*, pages 435–438.

Paul Ekman. 1992. An argument for basic emotions. *Cogition and emotion*, 6(3-4):169–200.

Xin Geng and Rongzi Ji. 2013. Label distribution learning. In *Proceedings of the 13th IEEE International Conference on Data Mining Workshops*, pages 377–383.

Xin Geng. 2016. Label distribution learning. *IEEE Transactions on Knowledge and Data Engineering*, 28(7):1734–1748.

Narendra Gupta, Mazin Gilbert, and Giuseppe Di Fabbrizio. 2013. Emotion detection in email customer care. *Computational Intelligence*, 29(3):489–505.

Jacob Harris. 2011. Word clouds considered harmful. *Nieman Journalism Lab*.

Sheng-Jun Huang and Zhi-Hua Zhou. 2012. Multi-label learning by exploiting label correlations locally. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence*, pages 949–955, Toronto, Canada.

Chenghua Lin and Yulan He. 2009. Joint sentiment/topic model for sentiment analysis. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, CIKM '09, pages 375–384, New York, NY, USA. ACM.

Gilad Mishne and Maarten de Rijke. 2006. Capturing global mood levels using blog posts. In *AAAI Sympo-*

sium on Computational Approaches to Analysing Weblogs (AAAI-CAAW), pages 145–152, June.

Jorge Nocedal and Stephen Wright. 2006. *Numerical optimization*. Springer Science & Business Media.

Robert Plutchik. 1980. A general psychoevolutionary theory of emotion. *Theories of emotion*, 1.

Robert Plutchik. 2001. An argument for basic emotions. *American Scientist*, 89(4):344–350.

S. Poria, A. Gelbukh, A. Hussain, N. Howard, D. Das, and S. Bandyopadhyay. 2013. Enhanced senticnet with affective labels for concept-based opinion mining. *Intelligent Systems, IEEE*, 28(2):31–38.

Changqin Quan and Fuji Ren. 2010. Sentence emotion analysis and recognition based on emotion words using ren-cecps. *International Journal of Advanced Intelligence*, 2(1):105–117.

Xiaojun Quan, Qifan Wang, Ying Zhang, Luo Si, and Liu Wenyin. 2015. Latent discriminative models for social emotion detection with emotional dependency. *ACM Trans. Inf. Syst.*, 34(1):2:1–2:19.

Jesse Read, Bernhard Pfahringer, Geoff Holmes, and Eibe Frank. 2011. Classifier chains for multi-label classification. *Machine Learning*, 85(3):333–359.

James A. Russell. 2003. Core affect and the psychological construction of emotion. *Psychological Review*, 110(1):145–172.

Richard Socher, Jeffrey Pennington, Eric H Huang, Andrew Y Ng, and Christopher D Manning. 2011. Semi-supervised recursive autoencoders for predicting sentiment distributions. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 151–161.

Carlo Strapparava and Alessandro Valitutti. 2004. Wordnet-affect: an affective extension of wordnet. In *Proceedings of the 4th International Conference on Language Resources and Evaluation*, pages 1083–1086.

Yichen Wang and Aditya Pal. 2015. Detecting emotions in social media: A constrained optimization approach. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence*, pages 996–1002.

Min-Ling Zhang and Zhi-Hua Zhou. 2006. Multilabel neural networks with applications to functional genomics and text categorization. *IEEE Transactions on Knowledge and Data Engineering*, 18(10):1338–1351.

Min-Ling Zhang and Zhi-Hua Zhou. 2014. A review on multi-label learning algorithms. *IEEE Transactions on Knowledge and Data Engineering*, 26(8):1819–1837.

Min-Ling Zhang. 2009. Ml-rbf: Rbf neural networks for multi-label learning. *Neural Processing Letters*, 29(2):61–74.

Min-Ling Zhang. 2011. Lift: Multi-label learning with label-specific features. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, pages 1609–1614, Barcelona, Spain.