Lexical and Learning-based Emotion Mining from Text

Ameneh Gholipour Shahraki, Osmar R. Zaïane

University of Alberta
Canada
{ameneh, zaiane}@ualberta.ca

Abstract. Emotion mining from text refers to the detection of people's emotions based on observations of their writings. In this work, we study the problem of text emotion classification. First, we collect and cleanse a corpus of Twitter messages that convey at least one of the targeted emotions, then, we propose several lexical and learning based methods to classify the emotion of test tweets and study the effect of different feature sets. Our experimental results show that a set of Naïve Bayes classifiers, each corresponding to one emotion, using unigrams as features, is the best-performing method for the task. In addition we test our approach on other datasets, Twitter, and formally written texts and show that our approach achieves higher accuracy, compared with state-of-the-art methods working on these corpora.

1 Introduction

Emotion mining refers to all areas of detecting, analyzing, and evaluating humans' feelings towards different events, issues, services, or any other interest. This field aims to mine emotions based on observations of people's actions that can be captured using their writings, facial expressions, speech, movements, etc. Here we focus only on text emotion mining, more specifically the task of fine-grained classification of existing emotion(s) conveyed by a text into one (or more) of a set of predefined emotions.

We define a set of 9 emotions and build classifiers to predict the emotion expressed in text. Our work is based on P. Ekman's model of basic emotions [1] as well as P. Shaver's [2], which later was explored more by W. G. Parrott [3]. Ekman states that there are 6 basic emotions: *anger, disgust, fear, joy, sadness*, and *surprise*. Shaver and Parrott suggest the same basic set of emotions with the exception of removing *disgust* and adding *love*. We merge the two models and also add *thankfulness* and *guilt*. *Guilt* is known as a basic emotion by some psychologists such as C. E. Izard [4] but is not included in neither Shaver's nor Ekman's. We embrace *guilt* in our model because detecting it helps psychologists determine if a patient faces or will face depression or stress. Indeed, our application is in the context of mental health and particularly detecting depression in text messages is crucial.

For this work we first collect and assemble a corpus of 27,000 emotional tweets containing a number of samples from 9 basic emotions we then create, based on our Twitter corpus, a lexicon containing about 24,000 words for emotion mining research, each associated with a vector of weights corresponding to the 9 basic emotions in our model. Finally we experiment with several lexical and learning-based methods for classifying emotions on Twitter corpus as well as some other existing datasets, resulting in outperforming some of the state-of-the-art done on these datasets.

2 Related Work

There are theories that define emotion and suggest some sets of basic emotions. However, there are still some controversial issues regarding whether some particular human states are classified as an emotion or not, and there is no systematic agreements between the suggested models in the litterature. Scientific studies on classification of human emotions date back to 1960s. There are two prevalent theories in this field. The first one, *discrete emotion theory*, states that different emotions arise from separate neural systems. Conversely, *dimensional model* states that a common and interconnected neurophysiological system is responsible for all affective states. This model defines emotions according to one or more dimensions where usually one of them relates to intensity of emotions. Basic emotions refer to those that do not have any other emotion as constituent parts. Many theorists on both sides have proposed sets of emotions that tend to be basic ones. Ekman, one of the earliest emotion theorists, suggested that those certain emotions that are universally recognized form the set of basic emotions (*anger*, *surprise*, *joy*, *disgust*, *fear*, and *sadness*). He later expanded his set of emotions by adding 12 new positive and negative emotions [5]. The dimensional model of R. Plutchik and H. Kellerman [6] arranges emotions in four bipolar axes: *joy* vs. *sadness*; *anger* vs. *fear*; *trust* vs. *disgust*; and *surprise* vs. *anticipation.* The fact that some of these emotions are actually opposite of each other is trivial in cases like *joy* vs. *sadness* but it is less intuitive in other cases, such as *anger* vs. *fear.* P. Shaver et al. [2] model emotions in a tree structure such that basic emotions are the main branches and each branch has its own categorization. H. Lövheim also suggests a dimensional model; however, his model is different from Plutchik's [7]. He believes that three hormones of serotonin, dopamine, and noradrenaline form three dimensions of a cube, where each basic emotion is placed on one of the corners.

There are two general approaches to the problem of textual emotion mining: A *lexical-based* method exploits a lexicon of words to decide about emotions of each or a group of words in a text and then aggregates those information to predict the total emotion of the whole document. A *learning-based* method applies some machine learning algorithms on a set of training data, in order to be able to predict the emotion of unseen test data. A lexicon may still be used to help doing feature selection or extraction.

With learning-based methods, the algorithm is provided with training data manually labeled with the emotion of each sample, which is expensive.

3 Creating a cleaned balanced emotional tweet (CBET) dataset

Twitter, with its millions of active users, reflects daily thoughts and concerns of people beyond compare. While the data is publically available, it promises a wider diversity of users. Unfortunately, to the best of our knowledge there are only three datasets from English tweets, available for public use where emotion expression is labeled [8–10]. Each of these corpora have drawbacks that make them open to criticism for being used in emotion mining research. W. Wang et al. [8] use keywords that are not really reflecting the proper emotion, such as the use of the hashtag #embarras as a clue for tweets having the emotion *sadness*. Mohammad's dataset, TEC [9], is imbalanced and labeling in Hasan's dataset [10] is based on a very different model of emotions which has only two dimensions of active-inactive and happy-unhappy. Therefore, it seems a new dataset is needed to overcome the drawbacks of previous ones. Previous research has shown that hashtags serve as acceptable emotion labels for tweets [8, 10]. Therefore, we decide to use this finding by searching for tweets with emotional hashtags and use those hashtags as tweets' labels.

Table 1. Hashtags used to search for tweets

Emotion	**List of Hashtags**
anger	#anger, #angry, #rage
fear	#fear
joy	#happy
love	#love
sadness	#sad
surprise	#surprise
thankfulness	#thankful
disgust	#disgust, #disgusted, #disgusting
guilt	#guilty, #sorry

Table 1 shows the corresponding hashtags that we use to retrieve our tweets of each emotion. According to this table, in the cases of *anger*, *disgust*, and *guilt* more than one hashtag is used to retrieve emotional tweets. The reason is that the number of tweets fetched using only one hashtag was not sufficient and would make the dataset imbalanced, so we added more hashtags by making very slight variations in order to take more tweets. A ballanced collection is desired to build a more robust and even lexicon. The hashtags #anger, #fear, #love, #surprise, and #disgust are identical to the name of their corresponding emotions. However, we use #happy over #joy for the emotion *joy* because it is a more informal and common word for describing joy on Twitter. The same reason applies for #thankful over #thankfulness for *thankfulness* and #sad over #sadness for *sadness*.

A total of $208,544$ tweets were initially collected in a time frame of 4 weeks from Oct. 31^{st} 2014 to Nov. 27^{th} 2014. Tweets of this corpus do not belong to any specific domain and form a general-purpose dataset suitable for analysis of people's day-to-day use of Twitter. $96,048$ duplicate tweets were detected and removed. We used a language detection library [11] which has the precision of over 99% to remove $21,599$ non-English tweets. $1,691$ tweets that contain 5 mentions or more (mentioning other users, with the pattern of "@" followed by a username) were removed. All other mentions were changed to a unified form, @user. $1,121$ less than 3 word tweets were also removed. For identical tweets with a Dice similarity > 0.3 only one is kept omitting $6,900$ tweets. The remaining $76,860$ tweets were further processed to remove the hashtags that served as the label, convert the capital letters to small ones, remove all URLs, stop words, numbers, useless punctuation marks, and redundant white spaces, and expand the space-free phrase hashtags to their constituent words. For instance #animalrights is expanded to *animal*, *rights* and the original hashtag. Finally, the remaining tweets were tokenized. we selected $3,000$ samples (tweets) of each emotion which gives us a dataset with total of $27,000$ samples which we refer to as *Cleaned Balanced Emotional Tweets (CBET)*. CBET is publically available at http://www.cs.ualberta.ca/~zaiane/CBET/.

4 Emotion Classification on CBET

We first introduce a lexical-based approach toward predicting the emotion of a writer and then explore several settings of learning-based methods. In particular, feature selection, dimension reduction, different configurations and learning algorithms are investigated.

4.1 Lexical-based classification

One of the very widely used approaches toward the problem is the lexical-based method. The simple intuition behind this technique is to look for emotional clues inside the text. In these approaches, one or more external resources are exploited for classification. Most frequently, these resources are in the form of lexicons that contain information about the emotion(s) or at least the polarity that words or phrases convey. Having such lexicons, the content of a message is evaluated based on the emotion(s) that its words or phrases have and a decision is made based on this information. In the problem of working with tweets, most of the existing lexicons are not suitable to use due to heavy load of abbreviations and informal language used in them. Therefore, we built an emotion lexicon from our Twitter corpus. The idea of developing this emotion lexicon is adopted from [12]. More concretely, dividing the corpus into training and test sets, we inspect the training set S word by word to see which words express which emotions and to what degree. For this purpose, we build a lexicon from the vocabulary V of all the single words (unigrams) contained in S. The lexicon is actually a $V \times E$ matrix where the element at index (j, i) denotes the degree that the word w_j expresses emotion e_i. In other words, each word has a corresponding weight vector that contains weights associated to each of the 9 basic emotions. The weight $F(e_i|w_j)$ is calculated as the number of times that w_j has occurred in tweets that have label e_i in the training set. That is:

$$F(e_i|w_j) = \sum_{s \in S} F(e_i|s) \times I_s(w_j) \tag{1}$$

where $F(e_i|s)$ is the presence of emotion e_i given sample s and $I_s(x)$ is an *indicator function* which is equal to 1 if $x \in s$ and is 0 otherwise.

The naïve assumption supporting this idea is that all the words in a tweet are in agreement with the label of that tweet. For example, if the training set contains *"Today is my birthday"* with label *joy*, *"I just forgot my mother's birthday"* with label *sadness*, and *"Hey! I was invited to her birthday!"* with label *joy*, then the weight vector for word *birthday* would be $\{0, 0, 2, 0, 1, 0, 0, 0, 0\}$ where index 3 and 5 are corresponding to *joy* and *sadness* respectively.

When classifying a new tweet, the weight vectors of its unigrams in the previously built lexicon are looked up and aggregated. The emotion that has the maximum aggregated weight would be the predicted label for the tweet, if we want a single label per tweet. Table 2 indicates the precision, recall, and F1 measure values, all in percent, after executing the lexical method on the corpus. All results are averages of 5 independent runs of the experiment. In each run, the corpus is shuffled and then 75% of tweets are randomly selected to be the training samples and the remained 25% form the test set. As the table shows, the average F1 measure for all emotions is 40.50% which is a great improvement over the baseline with random labelling (1/9=11.11%). *Thankfulness, love,* and *fear* are the easiest emotions to predict while *sadness* and *anger* are the hardest.

Table 2. Results of running lexical method

Emotion	**P**	**R**	**F1**
anger	40.28	24.10	30.10
fear	55.96	39.48	46.27
joy	46.88	35.52	40.39
love	51.50	43.58	47.17
sadness	30.69	24.54	27.26
surprise	48.00	34.62	40.20
thankfulness	42.36	57.26	48.64
disgust	43.50	30.34	35.73
guilt	23.14	58.86	33.16
ALL	**42.48**	**38.70**	**40.50**

The lexical-based method is easy and fast to build; however, it has some major drawbacks listed below:

1. If an external lexicon is to be used, it is hard to obtain and often domain-specific. If the lexicon is built from the working dataset, the model may not be reusable for other datasets and the process should be repeated for a new collection.
2. Even if the lexicon and training data are taken from the same domain, some words have different meanings in different sentences. For example, "*I had a great time with my grandfather today*" and "*My great-grandfather passed away yesterday*" show different meanings of "*great*".
3. Syntax structure of sentences can also influence the interpretation of words even if the meaning is clear. For instance, "*I laughed at him*" and "*He laughed at me*" differ only in the order of words, nevertheless, they most probably have different emotions from writer's point of view. These linguistic information are not usually included in normal lexicons and should be added in the form of an ontology [13].

4.2 Learning-based classification

Machine learning approaches have shown very good results in sentiment classification of text messages. These methods essentially try to learn patterns from a training set in which messages are labeled and then these patterns are used to guess the label of some new messages that the algorithm has not seen before. Considering the capabilities of the binary SVM, we decide to use it as the learning algorithm for our task. In order to have a 9-class classifier, we train 9 SVM classifiers, one for each emotion. The emotion that has the highest probability among all 9 emotions is predicted as the label of the test tweet. For an SVM responsible for learning emotion i, there are 3,000 positive samples (i.e. tweets with label i) and 3,000 negative samples (i.e. those with any label other than i). Negative samples are selected using an undersampling process. To do the undersampling, the negative samples are randomly permuted and then the first 3,000 ones are selected. Selecting features that distinguish samples of different classes plays an important role in the performance of SVM. We experiment several configurations of features. The most straight-forward way to come up with a set of features, is to use a lexicon and represent each tweet by a binary vector such that element i in the vector is 1 if the message has the word i from the lexicon and is 0 otherwise. This way, the features are those constituent words (unigrams) of text that exist in that lexicon and the set of all features is called "bag of words" since the words that are used in the training samples are considered but their order in making sentences is ignored. This method, generates presence-based features as it only keeps information about presence or absence of words (features) in binary format. Another alternative, namely frequency-based features, captures how many times each word is occurring in the text and the value of the features are thus positive integer values, instead of binary. Our results did not show any improvement when using frequency-based features, so we stick to presence-based representation of samples. Therefore, the input to the SVM algorithm can be seen as a $S \times V$ matrix, where each row is a training sample and each column is a vocabulary word taken from a lexicon. We run 4 experiments each with a different lexicon suitable for the emotion detection task: LIWC [14], NRC [15], NRC-hashtag [9] and our CBET lexicon. If a word from the lexicon is not used in at least 3 of the training tweets, that word is removed from the feature set. The reason is that such words very much enlarge the feature space but very rarely contribute to describing the messages. The feature set sizes after removing rare words are 440, 600,1300 and 3000 when exploiting LIWC, NRC, NRC-hashtag and our CBET lexicon. Other lexicons such as Wordnet Affect and WPARD have a much smaller feature set and will probably produce poor results over *CBET*. The best result (averages after 5 independent runs with cross validation), depicted in Table 3 was obtained with the our CBET lexicon followed by NRC-hashtag lexicon (**P=45.01**, **R=42.26**, **F1=43.59**).

Table 3. Results of SVM with words from CBET lexicon

Emotion	P	R	F1
anger	39.92	36.50	38.08
fear	56.61	57.31	56.95
joy	48.30	46.73	47.48
love	55.51	52.82	54.07
sadness	36.45	28.87	32.19
surprise	48.09	45.17	46.57
thankfulness	58.32	59.07	58.66
disgust	41.20	51.14	45.62
guilt	39.22	45.72	42.14
ALL	**47.07**	**47.04**	**47.05**

Moreover, we experimented with additional features. Instead of using all the non-rare unigrams as useful features, we imposed a criterion on them to discriminate useful unigrams from misleading ones. We define the notion of *informativeness* as a measure to see how informative a word is. Here, the concept of informativeness is close to *support* and *confidence* from association rule mining [16]. It includes both how frequent the word is and how much useful information it provides. The informativeness is calculated using a lexical-based approach. Suppose we have n tweets and using the leave-one-out method, classify each of them based on a lexicon that is built from the other $n-1$ ones and we do this n times so that all tweets are classified. Thus, for a unigram u, the informativeness, t_u is defined as:

$$t_u = \frac{CorrectClassify(u)}{TotalClassify(u)} \tag{2}$$

where $TotalClassify(u)$ shows the total number of times that u is used for classification and $CorrectClassify(u)$ is the number of times that we classify a tweet containing u correctly, if we solely use the weight vector of u for classification. In other words, $CorrectClassify(u)$ is an indicator of how much the emotion(s) coupled with u are consistent with the total emotion conveyed by the test tweet. The informativeness value ranges between [0,1] where 0 means the word is not informative at all and 1 shows the word is perfectly informative. We then filter the unigrams based on their informativeness value $t_u \geq 0.5$.

One of the key features of informal texts is the use of emoticons. Emoticons are easy to use and universally understandable symbols that are embedded in text and portray a wide range of emotions and hence are helpful resources for our problem. Particularly in *CBET*, more than 3% of the samples have at least one emoticon. The most frequently used ones are :), :(, :D, and ;) that respectively form 34%, 16%, 8%, and 8% of the whole emoticons used in the corpus. We expect that these emoticons would considerably help in emotion detection. Hence, in addition to unigrams, we add some boolean features, one for each of a set of 96 most used emoticons, representing the existence of that emoticon in the tweet. We experimented with informative unigrams then added the emoticons and the best result was with the combination including emoticons. The average results of 5 independent runs are shown in Table 4, where in addition to the informativeness filter, the emoticon features are also added in building the classifiers. Exploiting emoticons leads to only a slight improvement on precision, recall, and hence F1, which might not be inline with what we expect of emoticons. The reason could be in the method of using them. Further work on coming up with other methods of employing emoticons may lead to more substantial improvements. One suggestion is to use emoticons as final discriminator between two or more labels if their generated probabilities by SVM classifiers are so close that making a decision between them is hard for the system. In such cases, the existence of an emoticon in the test tweet may help to decide in favor of the correct label. Note that using PCA to reduce the unigram featureset from 3000 to 2000 significantly hindered the performance. In addition to emoticons, which are character-based, people also use emojis in their messages. These are ideograms, pictoral representations of objects or emotions. We did not consider these emojis or pictoral Smileys in our studies. They were considered as images and ignored.

In the literature, SVM is almost always the method of choice for learning-based sentiment analysis. We have investigated the same model as above but using Naïve Bayes, training a set of 9 Naïve Bayes classifiers on the training data, one for each emotion. Each classifier is fed with a balanced training set consisting of 3,000 positive (expressing that emotion) and 3,000 negative (expressing other emotions) instances. The set of features is taken directly from the vocabulary of the training tweets with rare words removed. The features used to train the Naïve Bayes method are informative unigrams plus the vector of 96 emoticons. The results are demonstrated in Table 5. Naïve Bayes has the average F1 value of 49.49% which is the best, compared to all previous methods. The standard deviations of 5 runs for precision, recall, and F1 are very low which is a proof of stability of the method. Interestingly, the best predictive power is achieved

Table 4. Results of SVM with informative words & emoticons

Emotion	P	R	F1
anger	39.10	38.74	38.90
fear	57.24	58.21	57.70
joy	48.90	44.77	46.74
love	54.88	54.02	54.41
sadness	33.89	31.36	32.55
surprise	49.21	45.62	47.33
thankfulness	58.31	58.38	58.32
disgust	42.29	48.92	45.31
guilt	41.25	44.56	42.79
ALL	**47.23**	**47.17**	**47.20**

with the *fear* samples, which might mean that people describe their fear feelings clearly and without mixing with other emotions.

Table 5. Results of running Naïve Bayes

Emotion	P	R	F1
anger	46.55	35.89	40.48
fear	62.01	58.58	60.22
joy	50.25	47.75	48.95
love	71.77	39.66	51.07
sadness	37.71	33.53	35.46
surprise	45.73	52.31	48.78
thankfulness	53.33	65.79	59.01
disgust	47.60	51.48	49.43
guilt	37.24	53.57	43.78
ALL	**50.27**	**48.73**	**49.49**

The confusion matrix resulting from the Naïve Bayes method is shown in Table 6. Here, in addition to conflicts in pairs of *sadness-guilt* and *sadness-disgust*, observed previously, the pairs *disgust-guilt*, *anger-guilt*, and *joy-surprise* show a high confusion as well. However, one of the very intertwined pairs of emotions, i.e. *joy-love*, is managed better in the Naïve Bayes classifier (31.4 confused cases) rather than SVM (99.2 confused cases). According to the table, positive emotions such as *love* and negative ones such as *guilt* or *disgust* are the most separable labels. In Table 6 , the number at row i and column j is the number of samples that have true label i but are predicted to have label j. Note that each number is the average of results of 5 independent runs. An ideal classifier should have non-zero numbers on the diagonbal and all the numbers on non-diagonal positions equal to 0.

Table 6. Confusion matrix for the Naïve Bayes model

Emotion	anger	fear	joy	love	sad	surprise	thankful	disgust	guilt
anger	**260.6**	49.2	44.4	14	76.8	54.4	36.2	94.2	113.2
fear	34.2	**455.6**	24.4	10.4	45.8	32	42.8	61.6	60.2
joy	29.8	27.8	**353**	31.4	37.8	112.4	92	12.8	45.2
love	25.2	53.4	107.2	**305.6**	35.2	81	79.2	22.4	51.2
sad	71.2	36.6	34.4	14	**245**	47.2	52.4	103.8	139
surprise	36.4	21.4	63.2	14	45.6	**384**	65.4	30	86.4
thankful	19.8	22.8	37	8.2	40	64.8	**487.4**	25	46.4
disgust	55.8	25.4	12.4	4	73.8	39.6	33	**391.8**	114.8
guilt	45	22.8	20.6	8.8	92.2	51.6	42.6	73.6	**388.2**

5 Emotion Classification on other datasets

Our proposed methods achieved acceptable results over our Twitter corpus, *CBET*; nevertheless, it is intersting to assess these on other datasets and compare the results with state-of-the-art methods. For this purpose, in what follows we test our methods on another Twitter dataset and a dataset of formal documents.

Twitter Emotion Corpus (TEC) is collected by S. M. Mohammad [9] in 2012. Mohammad targets 6 basic emotions: *anger, disgust, fear, joy, sadness,* and *surprise* and searches for tweets having a hashtag corresponding to one of these emotions. After pre-processing, *TEC* includes 21,051 tweets where 7.4%, 3.6%, 13.4%, 39.1%, 18.2%, and 18.3% of the corpus have the aforementioned emotions, respectively, which shows the corpus is imbalanced. In order to address the emotion classification problem, the author builds 6 binary SVM models with Sequential Minimal Optimization [17], one for each emotion, using unigrams and bigrams as features. When classifying an unseen tweet, for each emotion the corresponding classifier is applied to decide whether the tweet has that emotion or not. This way, a tweet may get zero, one or multiple labels. Precision, recall, and F1 value of this method is shown in Table 7. According to this table, *joy* and *disgust* have the best and worst prediction results, respectively. However, the effect of the size of training samples for each emotion should not be neglected. The better results for *joy* may be due to the large number of training tweets labeled as *joy* (39.1% of the dataset). It seems that there exists a correlation between the size of the training set and the performance, such that, if the number of training samples increases, then the system achieves a better F1 value. This is inline with the findings of W. Wang et al. [8] who suggest that "*learning from large training data can play an important role in emotion identification*". The only exception in *TEC* is for the emotion *fear*, that in spite of fewer training samples than *sadness* and *surprise* achieves a better performance.

Similar to Mohammad, we trained 6 binary SVM classifiers using unigrams and bigrams. However, in the classification phase we use our method where each classifier outputs a probability for a given test tweet and the emotion showing the maximum probability is the predicted label. Except for this labeling procedure, all stages are kept as Mohammad's. The results show a remarkable improvement particularly for the recall (**P=43.88**, **R=46.90**, **F1=45.34**). However, when removing the bigrams there is no major difference with our results including them. Finally, when replacing the 6 SVMs with 6 Naïve Bayes using only unigram features, results are even better (Table 8).

Table 7. Results of Mohammad approach on TEC

Emotion	P	R	F1
anger	37.3	22.31	27.9
fear	59.6	43.9	50.6
joy	64.5	60.4	62.4
sadness	41.9	36.0	38.7
surprise	50.6	40.5	45.0
disgust	30.7	13.4	18.7
ALL	**47.4**	**36.1**	**40.98**

Table 8. Results of our approach on TEC

Emotion	P	R	F1
anger	30.37	45.22	36.29
fear	63.41	50.31	56.06
joy	71.99	69.08	70.49
sadness	47.00	51.71	49.21
surprise	62.60	40.28	48.94
disgust	17.08	42.67	24.29
ALL	**48.74**	**49.88**	**49.30**

One of the oldest emotion labeled datasets, freely available, is ISEAR [18]. The data was collected during 1990s, by a group of international psychologists. In this survey, $3,000$ students, both psychologists and non-psychologists, in 37 countries on all 5 continents were asked to report situations in which they had experienced 7 major emotions: *joy, fear, anger, sadness, disgust, shame,* and *guilt.* This dataset is reliable in terms of labeling, since the authors, themselves, have annotated their text. However, translating from other languages to English might change the senses and emotions. Surprisingly, ISEAR was not used for emotion mining purposes until 2008. In the literature, there are some works done on emotion detection from *ISEAR.* The work by D. T. Ho and T. H. Cao [19] uses a high-order Hidden Markov Model (HMM) to address the problem. They take into account only *anger, fear, joy,* and *sadness* emotions where *anger* covers both *anger* and *disgust.* The best reported value for F1, averaged over 4 emotions, is 35.3% for the configuration of a 2nd-order HMM with 45 states trained on 2/3 of the samples and tested on the rest. S. M. Kim et al. [20] target the *ISEAR* dataset as well. They build 4 types of classifiers: discrete classifiers with LSA, PLSA, and NMF dimension reduction methods and a dimensional classifier. Similar to [19], they also consider *anger + disgust, fear, joy,* and *sadness.* The reported F1 values averaged for all emotions are 22.77%, 26.95%, 16.55%, and 37.22% for each of the mentioned classifiers, respectively. Note that since they consider 5 emotions, a random classifier acting as a baseline has F1 value of 1/5 = 20%. We tested our lexical-based method and the Naïve Bayes model on *ISEAR.* Tables 9 and 10 present the results. F1 values of 48.95% and 54.78% for two models show a significant improvement. To make the comparison fairer with previous works, we also consider only those 5 emotions suggested by them. The F1 values averaged over the 5 emotions for the lexical and the Naïve Bayes method are 50.97% and 55.94% which is significantly higher than both previous attempts. *Fear* and *joy* are the best predictable emotions while *anger* is the hardest among all. It seems that the ability of the classifiers to predict a specific emotion varies highly from one dataset to another. For instance, *sadness* is one of the toughest emotions to predict in *CBET* while it is predicted with the pretty good result in *ISEAR.*

Table 9. Running lexical method on ISEAR

Emotion	**P**	**R**	**F1**
anger	33.60	41.71	36.98
fear	57.29	56.41	56.74
joy	63.86	51.49	56.82
sadness	59.44	48.42	53.25
disgust	63.12	37.72	47.14
guilt	27.94	57.49	37.58
shame	58.95	30.26	39.80
ALL	**52.03**	**46.21**	**48.95**
5 emotions	**55.46**	**47.15**	**50.97**

Table 10. Running Naïve Bayes on ISEAR

Emotion	**P**	**R**	**F1**
anger	45.83	38.24	41.64
fear	65.68	64.28	64.92
joy	59.05	74.15	65.71
sadness	55.91	59.14	57.45
disgust	55.97	55.53	55.66
guilt	48.17	43.17	45.44
shame	51.25	50.50	50.85
ALL	**54.55**	**55.00**	**54.78**
5 emotions	**56.49**	**55.40**	**55.94**

6 Conclusion

We addressed the problem of text-based emotion classification. Personal notes, emails, news headlines, blogs, and chat messages are some types of text that can convey emotions. Particularly, popular social networking websites such as Twitter, Facebook, and MySpace are common places to share one's feelings. Emotion classification is an interesting topic in many disciplines such as neuroscience, cognitive sciences, psychology, and computer science and has many applications including e-learning systems, human-computer interaction, customer care services, and psychological cognition. To address the emotion classification problem, we first compiled a corpus of 27,000 emotional tweets, called *CBET*, that contains a balanced number of samples from 9 basic emotions: *anger, fear, disgust, joy, love, sadness, surprise, thankfulness,* and *guilt*. Next, we proposed a lexical-based method that basically evaluates the content of a message regarding the emotion(s) that its words or phrases have and a decision is made based on this information. In addition, several learning-based methods were suggested. They essentially try to learn patterns from a training set in which messages are labeled and then these patterns are used to guess the label of some new messages that the algorithm has not seen before. Also, the effects of different feature selection methods, dimension reduction approaches, other configurations of classifiers, and various learning algorithms were investigated. Our methods showed promising results over *CBET*. Additionally, they were shown to be capable with domain-independent performance such as being used for other Twitter and non-Twitter domains including *TEC* and *ISEAR*.

References

1. P. Ekman, W. V. Friesen, and P. Ellsworth, *Emotion in the human face: Guidelines for research and an integration of findings.* Pergamon Press, 1972.
2. P. Shaver, J. Schwartz, D. Kirson, and C. O'connor, "Emotion knowledge: further exploration of a prototype approach.," *Journal of personality and social psychology*, vol. 52, no. 6, p. 1061, 1987.
3. W. G. Parrott, *Emotions in social psychology: Essential readings.* Psychology Press, 2001.
4. C. E. Izard, *The psychology of emotions.* Springer Science & Business Media, 1991.
5. P. Ekman, "An argument for basic emotions," *Cognition & emotion*, vol. 6, no. 3, pp. 169–200, 1992.
6. R. Plutchik and H. Kellerman, *Emotion: theory, research and experience.* Academic press New York, NY, 1986.
7. H. Lövheim, "A new three-dimensional model for emotions and monoamine neurotransmitters," *Medical hypotheses*, vol. 78, no. 2, pp. 341–348, 2012.
8. W. Wang, L. Chen, K. Thirunarayan, and A. P. Sheth, "Harnessing twitter "big data" for automatic emotion identification," in *2012 International Conference on Privacy, Security, Risk and Trust (PASSAT) and 2012 International Confernece on Social Computing (SocialCom)*, pp. 587–592, IEEE, 2012.
9. S. M. Mohammad, "#emotional tweets," in *Proceedings of the First Joint Conference on Lexical and Computational Semantics (*SEM)*, pp. 246–255, Association for Computational Linguistics, 2012.

10. M. Hasan, E. Agu, and E. Rundensteiner, “Using hashtags as labels for supervised learning of emotions in twitter messages,” *health informatics workshop (HI-KDD)*, 2014.
11. N. Shuyo, “Language detection library for java.” `http://code.google.com/p/language-detection/`, 2010.
12. P. Katz, M. Singleton, and R. Wicentowski, “Swat-mp: the semeval-2007 systems for task 5 and task 14,” in *Proceedings of the 4th international workshop on semantic evaluations*, pp. 308–313, Association for Computational Linguistics, 2007.
13. E.-C. Kao, C.-C. Liu, T.-H. Yang, C.-T. Hsieh, and V.-W. Soo, “Towards text-based emotion detection a survey and possible improvements,” in *International Conference on Information Management and Engineering (ICIME)*, pp. 70–74, IEEE, 2009.
14. J. W. Pennebaker, M. E. Francis, and R. J. Booth, “Linguistic inquiry and word count: Liwc 2001,” *Mahway: Lawrence Erlbaum Associates*, vol. 71, p. 2001, 2001.
15. S. M. Mohammad and P. D. Turney, “Emotions evoked by common words and phrases: Using mechanical turk to create an emotion lexicon,” in *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pp. 26–34, Association for Computational Linguistics, 2010.
16. B. Liu, W. Hsu, and Y. Ma, “Integrating classification and association rule mining,” in *Proceedings of the fourth international conference on knowledge discovery and data mining*, 1998.
17. J. C. Platt *et al.*, “Using analytic qp and sparseness to speed training of support vector machines,” *Advances in neural information processing systems*, pp. 557–563, 1999.
18. K. R. Scherer and H. G. Wallbott, “Evidence for universality and cultural variation of differential emotion response patterning.,” *Journal of personality and social psychology*, vol. 66, no. 2, p. 310, 1994.
19. D. T. Ho and T. H. Cao, “A high-order hidden markov model for emotion detection from textual data,” in *Knowledge Management and Acquisition for Intelligent Systems*, pp. 94–105, Springer, 2012.
20. S. M. Kim, A. Valitutti, and R. A. Calvo, “Evaluation of unsupervised emotion models to textual affect recognition,” in *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pp. 62–70, Association for Computational Linguistics, 2010.