# Actionable Pattern Discovery for Tweet Emotions

Angelina Tzacheva(✉), Jaishree Ranganathan,
and Sai Yesawy Mylavarapu

Department of Computer Science, University of North Carolina at Charlotte,
Charlotte, NC, USA
{aatzache, jranganl, smylaval}@uncc.edu

**Abstract.** The most popular form of communication over the internet is text. There are wide range of services that allow users to communicate in the natural language using text messages. Twitter is one such popular Micro-blogging platform where users post their thoughts, feeling or opinion on a day-to-day basis. These text messages not only contain information about events, products and others but also the writer's attitude. This kind of text data is useful to develop systems, which detect user emotions. Emotion detection has wide variety of applications including customer service, public policy making, education, future technology, and psychotherapy. In this work, we use Support Vector Machine classifier model to automatically classify user emotions. We achieve accuracy in the range of 88%. The Emotional information mined from such data is huge and these findings can be more useful if the system is able to provide some actionable recommendations to the user, which help them, achieve their goal and gain benefits. The recommendations or patterns are Actionable if user can perform action using the patterns to their advantage. Action Rules help discover ways to reclassify objects with respect to a specific target, which the user intends to change for their benefits. In this work, we focus on extracting Action Rules with respect to the Emotion class from user tweets. We discover actionable recommendations, which suggests ways to alter the user's emotion to a better or more positive state.

**Keywords:** Actionable pattern discovery · Data mining · Emotion mining · Support Vector Machine · Scalability

## 1 Introduction

According to Merriam Webster, dictionary [1] Micro-blogging is blogging done with severe space or size constraints typically by posting frequent brief messages about personal activities. There are wide variety of such micro-blog services available on the web including Twitter [2], Tumblr [3], Pownce (http://pownce.com) and many others. Among these, Twitter is the most popular. According to ComScore [4], within eight months of its launch, Twitter had about 94,000 users as of April 2007 [5]. In addition, micro-blogging users may post several updates on a single day [5]. Approximately 500 million tweets are posted on Twitter per day. Thus, the amount of textual data generated is huge when we consider the rate of growth of Twitter user's since 2007 and the periodicity

T. Ahram (Ed.): AHFE 2019, AISC 965, pp. 1–13, 2019.
https://doi.org/10.1007/978-3-030-20454-9_5

of the posts on a single day by a user. It allows adding emoticons, which are one of the powerful tools to express human emotions. Hashtag is a tagging convention that helps people associate tweets with certain events or contexts [6]. It is a keyword prefixed with '#' symbol. These hashtags sometimes indicate the writer's emotion. For example the tweet, "Homemade chicken soup is the best #happy" indicates happiness [7].

Data mining from such rich sources of text helps gain useful insights in a range of applications. For instance, Gupta et al. [8] study the customer care email in- order to identify customer dissatisfaction and help improve business. Analyzing the social media posts of a particular community might help government officials in public policy making to improve the quality of life of people in that area. In educational domain, identifying student's thoughts and emotion about the university, faculty helps improve the quality of education. In the field of psychology, where online social therapy is used for assisting mental health as face-to-face early intervention services for psychosis is for limited time period and benefits may not persist after its termination [9] and in scenarios where machines are used as psychotherapist [10]. After information is gathered from such data, it is necessary to validate the mined information. For this purpose, there are many supervised learning models that help automatically classify new set of test data, given a considerable amount of data for training.

With the proliferation of information through various sources, there is access to enormous amount of data, at the same time leads to poor information in the raw form and inefficient decision-making [11]. The volume of discovered patterns is huge despite the use of data mining strategies, which leads to unreliable and uninteresting knowledge [11]. Actionable patterns are those that help users benefit by using it to their own advantage. Action Rules are special type of rules that help identify actionable patterns from the data [12].

## 2 Related Work

In this section, we review literature works in the areas of Emotion classification from text, and actionable pattern mining based on text classification.

### 2.1 Emotion Classification from Text

Mishne [13] classify writer's mood in blog text collected from Live Journal, a free weblog service using Yahoo API. They use following features to train the SVMlight model from Support Vector Machine package: frequency counts (words, Part-Of-Speech), and length of blog post; subjective nature of blogs like semantic orientation, Point-wise Mutual Information (PMI) which is a measure of the degree of association between two terms; features unique to online text like emphasized words, special symbols including punctuation's, and emoticons. They attribute subjective nature of the corpus "annotation" and nature of blog posts as major factors for low accuracy.

Danisman and Alpkocak [14] use Vector Space Model (VSM) where each document is a vector and terms correspond to dimensions and develop a text classifier. Term Frequency - Inverse Document Frequency (tf-idf) weighting scheme is used to calculate weight of each term in the document. They have analyzed the effect of emotional intensity and stemming to the classification performance. Results show that Vector Space Model performs equally well compared to other well-known classifiers.

Mohammad [15] developed corpus from Twitter posts using emotion hash-tags called Twitter Emotion Corpus (TEC) consisting of 21,000 tweets. Support Vector Machines (SVM) with Sequential Minimal Optimization (SMO) classifier was used with unigram and bigram features. The automatic classifiers obtained an F-score much higher than the random baseline (SemEval – 2007, 1000 headlines dataset). Similar to Wang et al., in this paper best results are achieved with higher number of training instances. For example, Joy-NotJoy classifier get the best results compared to Sadness-NotSadness.

Roberts et al. [16] create emotion corpus from Twitter. The corpus contains seven emotions annotated across 14 topics including Valentine's Day, World Cup 2010, Stock Market, Christmas etc. The emotions are based on Ekman's [17] six basic emotions and LOVE. The topics of each tweet obtained by considering the tweet associated with a probabilistic mixture of topics using Latent Dirichlet Allocation (LDA) topic modeling technique. The system uses a series of binary SVM classifiers to detect each of the seven emotions annotated in the corpus. Each classifier performs independently on a single emotion, resulting in 7 separate binary classifiers implemented using the software available from WEKA and uses specific set of features like punctuation, hypernyms, n-grams, and topics. According to the results, FEAR is the best performing emotion and suggests that this emotion is highly lexicalized with less variation than other emotions, as it has comparable recall but significantly higher precision. Overall, in this work, the macro-average precision is 0.721 and recall is 0.627.

Purver et al. [18] used Twitter data labeled with emoticons and hash-tags to train supervised classifiers. They used Support Vector Machines with linear kernel and unigram features for classification. Their method had better performance for emotions like happiness, sadness, and anger but not well in case of other emotions like fear, surprise, and disgust. They achieved accuracy in the range of 60%.

### 2.2 Actionable Pattern Mining

In [19] the primary intent of the Action Rules generated is to provide viable suggestions on how to make a twitter user feel more positive. For Twitter social network data, Actionable Recommendations include - how to increase user friend's count, and how to change the overall sentiment from negative to positive, or from neutral to positive.

## 3 Methodology

### 3.1 Data Collection

In data collection step we used Twitter streaming API [20] to collect the data with the following attributes TweetID, ReTweetCount, TweetFavouriteCount, TweetText, Tweet- Language, Latitude, Longitude, TweetSource, UserID, UserFollowersCount, UserFavoritesCount, UserFriendsCount, UserLanguage, UserLocation, UserTimeZone,

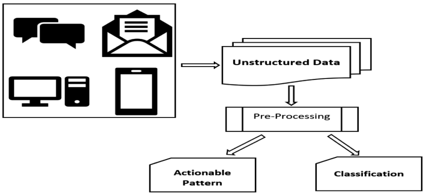


Fig. 1. Overall methodology.

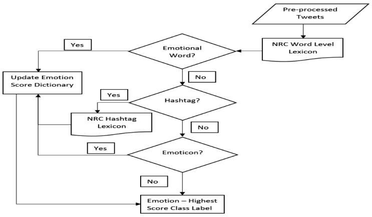


Fig. 2. Emotion labeling.

IsFavorited, IsPossiblysensitive, IsRetweeted, RetweetedStatus, UserStatus, MediaEntities. We collected around 520,000 tweets as raw data. Figure 1 shows the overall model of the proposed methodology.

### 3.2 Pre-processing

The extracted tweet text is pre-processed to make the informal text suitable for emotion classification. We lower case all the letters in the tweet; remove stop words i.e. the most frequent words in English which will not add value to the final emotion; replace slang words with formal text, example b4 → before, chk → check, etc. After pre-processing we have around 200,000 tweets.

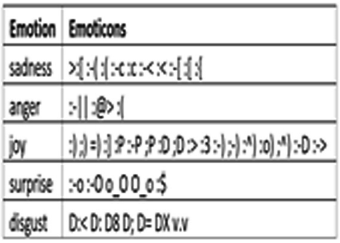
### 3.3 Emotion Labeling

To identify the emotion class, we use the National Research Council - NRC lexicon [21, 22]. The Annotations in the lexicon are at WORD-SENSE level. Each line has the format: <Term> <AffectCategory> <AssociationFlag> where Term is a word for which emotion associations are provided, Affect Category is one of the eight emotions anger, fear, anticipation, trust, surprise, sadness, joy, or disgust and one of two polarities negative or positive, Association flag indicates that the target word has association with category word or not.

Apart from word level annotation, to increase the weightage of each emotion class assigned to tweet we also use the hashtags and emoticons inside the tweet text. For hashtags, we utilize the NRC Hashtag Emotion Lexicon [15, 23], which is a list of words and their associations with eight emotions. The associations are computed from tweets with emotion-word hashtags such as #happy and #anger.

**Table 1.** Encoding categorical attributes.

| Attribute | Encoding |
|---|---|
| 'False':0, 'True':1 | 'iPhone':1, 'Android':2, 'TweetDeck':3, 'web':4 |
| UserLanguage | Each user language assigned a numeric value |
| MediaEntities | 'None':0, 'photo':1 |
| IsPossiblySensitive | |



| Emotion | Emoticons |
|---|---|
| sadness | >:[ :-( :( :-c :c :-< :< :-[ :[ :{ |
| anger | :-\|\| :@> :( |
| joy | :) ;) =) :] :P :-P ;P :D ;D >:3 :-) ;-) :^) :o) ;^) :-D :-> |
| surprise | :-o :-O o_O O_o :$ |
| disgust | D:< D: D8 D; D= DX v.v |

**Fig. 3.** Emoticons.

All emoticons retained in the data collection process and validated while assigning weights to each emotion class. Figure 3 shows the list of emoticons used in this process. Figure 2 explains the steps involved in assigning final emotion class.

### 3.4 Additional Pre-processing

In addition to pre-processing steps for text data, additional pre-processing (Table 1) is performed on the numeric attributes of data in order to make it suitable for Classification. The data set has the following Numeric Attributes: AngerScore, TrustScore, FearScore, Sad-nessScore, AnticipationScore, DisgustScore, SurpriseScore, JoyScore, PositiveScore, NegativeScore, LoveScore, PeopleScore, MessageScore, InstantScore, GetScore, KnowScore, GoingScore, UserFollowersCount, UserFavoritesCount, UserFriendsCount and are normalized using python scikit learn MinMaxScaler in the range of −1 to +1. After additional pre-processing the data is converted into LIBSVM [24] format that is suitable for classification using Support Vector Machine.

### 3.5 Emotion Classification

Classification is a supervised machine learning model that learns the data using labeled train set and predicts the test set for which the model does not know the actual class labels. This model is further evaluated with the help of validation measures like precision, recall, f1-score, and accuracy. In this paper, we have used Support Vector Machine as a classification model for automatically classifying Twitter dataset with emotion. Figure 4 shows the overall processing flow of Support Vector Machine classification utilized in this work.

Support Vector Machines – SVM: Support Vector Machines (SVM) are a useful technique for data classification originally designed for binary classification [25, 27]. Hsu and Lin [27] provide overview of methods for multiclass support vector machines. In order to extend binary SVM to multiclass problems there are three methods: ONE-AGAINST-ALL, ONE-AGAINST-ONE, and Directed Acyclic Graph SVM -DAGSVM methods. Formal definition of these methods as stated in [27]: ONE-AGAINST-ALL: This method constructs k SVM models, where k is the number of classes. The ith SVM is trained with all of the examples in the ith class with positive labels, and all other examples with negative labels. ONE-AGAINST-ONE: This method constructs k(k − 1)/2 classifiers where each one is trained on data from two

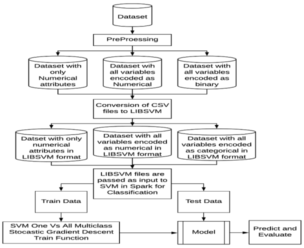
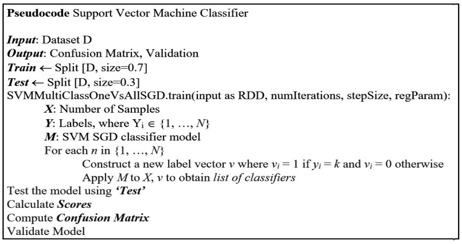

Dataset
PreProessing
Dataset with only Numerical attributes
Dataset wih all variables encoded as Numerical
Dataset with all variables encoded as binary
Conversion of CSV files to LIBSVM
Datset with only numerical attributes in LIBSVM format
Dataset with all variables encoded as numerical in LIBSVM format
Dataset with all variables encoded as categorical in LIBSVM format
LIBSVM files are passed as input to SVM in Spark for Classification
Train Data
Test Data
SVM One Vs All Multiclass Stocastic Gradient Descent Train Function
Model
Predict and Evaluate

**Fig. 4.** Process – Support Vector Machine.



```
Pseudocode Support Vector Machine Classifier

Input: Dataset D
Output: Confusion Matrix, Validation
Train ← Split [D, size=0.7]
Test ← Split [D, size=0.3]
SVMMultiClassOneVsAllSGD.train(input as RDD, numIterations, stepSize, regParam):
        X: Number of Samples
        Y: Labels, where $Y_i \in \{1, \ldots, N\}$
        M: SVM SGD classifier model
        For each $n$ in $\{1, \ldots, N\}$
                Construct a new label vector $v$ where $v_i = 1$ if $y_i = k$ and $v_i = 0$ otherwise
                Apply $M$ to $X$, $v$ to obtain list of classifiers
Test the model using 'Test'
Calculate Scores
Compute Confusion Matrix
Validate Model
```

**Fig. 5.** Support Vector Machine - Pseudocode

classes. Figure 8 shows sample implementation of ONE-AGAINST-ONE method. Directed Acyclic Graph SVM - DAGSVM: The training phase of DAGSVM is the same as one-against-one method by solving k(k − 1)/2 internal nodes and k leaves. Each node is a binary SVM of ith and jth classes. Given a test sample x, starting at the root node, the binary decision function is evaluated. Then it moves to either left or right depending on the output value. Therefore, we go through a path before reaching a leaf node, which indicates the predicted class. In this paper, we utilize the ONE-AGAINST-ALL method, pseudocode shown in Fig. 5.

### 3.6 Actionable Pattern Mining

Actionability is a property of the discovered knowledge. Patterns are considered Actionable if the user can act upon them, and if this action can benefit the user, or help them to accomplish their goals. Action Rules mining is a method to extract Actionable patterns from the data. Action Rules are rules that describe a possible transition of data from one state to another more desirable state. Action Rules are rules that help reclassify data from one category to another more desirable category. Consider the information system S in Table 2. Equations (1) and (2) are example Action Rules. According to Eq. (1) r1 says that if the value A2 remains unchanged and value B will change from B2 to B1 for a given object X, then it is expected that the value D will change from H to A for object X.

In a similar way, the rule r2 in Eq. (2) says that if the value C2 remains unchanged and value b will change from B2 to B1, then it is expected that the value D will change from H to A.

$$\mathrm{r1} = [((\mathrm{A}, \mathrm{A2} * (\mathrm{B}, \mathrm{B2} \rightarrow \mathrm{B1})) \rightarrow (\mathrm{D}, \mathrm{H} \rightarrow \mathrm{A})]. \tag{1}$$

$$\mathrm{r2} = [((\mathrm{C}, \mathrm{C2} * (\mathrm{B}, \mathrm{B2} \rightarrow \mathrm{B1})) \rightarrow (\mathrm{D}, \mathrm{H} \rightarrow \mathrm{A})]. \tag{2}$$

By support and confidence of rule r we mean:

1. sup(r) = min{card(Y1 ∩ Z1), card(Y2 ∩ Z2)}
2. conf(r) = card(Y1 ∩ Z1)/ card(Y1). card(Y2 ∩ Z2)/ card(Y2). If card(Y1) ≠ 0, card(Y2) ≠ 0, card(Y2 ∩ Z2) ≠ 0.
3. Conf(r) = 0 otherwise.

Now, let us consider the Eq. (1) for support and confidence with example.

- $N_s(a, a_2 \rightarrow a_2) = [\{x_2, x_3, x_5, x_6, x_{10}\}, \{x_2, x_3, x_5, x_6, x_{10}\}]$
- $N_s(b, b_2 \rightarrow b_1) = [\{x_2, x_6, x_8, x_{10}\}, \{x_1, x_3, x_4, x_5, x_7, x_9\}]$
- $N_s(a, a_2 \rightarrow a_2) * (b, b_2 \rightarrow b_1) = [\{x_2, x_6, x_{10}\}, \{x_3, x_5\}]$
- $N_s(d, H \rightarrow A) = [\{x_1, x_2, x_6, x_9, x_{10}\}, \{x_3, x_4, x_5, x_7, x_8\}]$

**Table 2.** Information system S.

| X | Attribute A | Attribute B | Attribute C | Attribute D |
|---|---|---|---|---|
| X1 | A1 | B1 | C1 | H |
| X2 | A2 | B2 | C1 | H |
| X3 | A2 | B1 | C1 | A |
| X4 | A1 | B1 | C2 | A |
| X5 | A2 | B1 | C2 | A |
| X6 | A2 | B2 | C2 | H |
| X7 | A1 | B1 | C2 | A |
| X8 | A1 | B2 | C1 | A |
| X9 | A1 | B1 | C1 | H |
| X10 | A2 | B2 | C1 | H |

Therefore, for rule r1, support sup(r1) = 2, confidence conf(r1) = 3/2 . 3/2 = 1.

Tzacheva et al. [34] describe that these formulas for support and confidence are too complex for computation provide definition of new support and confidence for Action Rules as below. New support and confidence of rule r is given as sup(r) = card($Y_2 \cap Z_2$), conf(r) = card($Y_2 \cap Z_2$)/ card($Y_{2)}$.

### 3.7 Spark Scalability for Big Data

Increase in data size and the need to scale out computations to multiple nodes gave rise to the distributed programming models. One such model is Apache Spark, which is similar to Hadoop MapReduce. In addition to the similarities, Apache Spark includes data sharing abstraction called Resilient Distributed Dataset's (RDD's) [30].

## 4 Experiments and Results

In this section, we describe our experiment and results. We extract the data via Twitter streaming API [20] using Apache Spark [31] Scala programming language. The raw data extracted consists of around 520,000 instances. The extracted data is processed as explained in Sect. 3.2. As part of feature, augmentation additional attributes are added to the existing corpus along with the emotion label. We use the NRC Lexicon [15, 23] to label data with Emotion Class as shown on Fig. 2. This results in a corpus of tweets and supporting features consisting of around 174,000 instances.

### 4.1 Classification – Support Vector Machine

We use WEKA Data Mining Software [32] and Apache Spark [31] to develop the Support Vector Machine One Vs All Multi class classifier. Support Vector Machine classification model requires pre-processing of data, which includes normalization, categorical to numeric or binary, LIBSVM format. Based on the pre-processing three experiments are performed: Using only the numerical attributes in the data, Using all the attributes where the categorical fields are encoded as numeric values, Using all attributes where the categorical fields are encoded as binary.

**Experiment 1 - Using Only Numeric Attributes:** Experiment 1 is performed by selecting only the numerical attributes from the original dataset which includes: AngerScore, TrustScore, FearScore, SadnessScore, AnticipationScore, DisgustScore, SurpriseScore, JoyScore, PositiveScore, NegativeScore, LoveScore, PeopleScore, MessageScore, InstantScore, GetScore, KnowScore, GoingScore, UserFollowersCount, UserFavoritesCount, User- FriendsCount.

We achieve accuracy of 84.92% with WEKA Data Mining software Multiclass Classifier with SVM Stochastic Gradient Descent (SGD) Tables 3 and 4.

**Table 3.** WEKA – confusion matrix – experiment 1.

| A | B | C | D | E | F | G | H | Class |
|---|---|---|---|---|---|---|---|---|
| 10133 | 0 | 0 | 1481 | 0 | 0 | 0 | 0 | A - Sadness |
| 144 | 5535 | 1 | 2080 | 0 | 0 | 57 | 0 | B - Joy |
| 152 | 6 | 1477 | 716 | 17 | 0 | 2 | 0 | C - Fear |
| 104 | 10 | 33 | 15150 | 3 | 0 | 4 | 0 | D - Anticipation |
| 127 | 42 | 22 | 339 | 6342 | 0 | 1 | 0 | E - Trust |
| 50 | 18 | 10 | 293 | 213 | 1124 | 0 | 1 | F - Surprise |
| 165 | 6 | 85 | 179 | 61 | 3 | 2600 | 0 | G - Anger |

**Table 4.** WEKA – precision, recall, F-measure – experiment 1.

| Measure | Sadness | Joy | Fear | Anticipation | Trust | Surprise | Anger | Disgust |
|---|---|---|---|---|---|---|---|---|
| Precision | 0.922 | 0.985 | 0.892 | 0.712 | 0.935 | 0.996 | 0.926 | 1.000 |
| Recall | 0.872 | 0.708 | 0.623 | 0.990 | 0.923 | 0.658 | 0.839 | 0.593 |
| F-Measure | 0.897 | 0.824 | 0.734 | 0.828 | 0.929 | 0.792 | 0.880 | 0.744 |

We achieved almost similar accuracy with Spark single node and six-node cluster as 88.16% and 88.01% respectively. The confusion matrix and classifier evaluation with precision, recall, and F1-score is shown in Tables 6, and 5 respectively.

However, the Spark program runs faster in cluster when compared to Single Node machine for all the three experiments. The results of average run time for execution is shown in Table 7.

**Table 5.** Spark – precision, recall, F-measure – experiment 1.

| Measure | Anticipation | Sadness | Joy | Trust | Disgust | Anger | Fear | Surprise |
|---|---|---|---|---|---|---|---|---|
| Precision | 0.8898 | 0.8313 | 0.8848 | 0.8912 | 0.9634 | 0.9248 | 0.9153 | 0.0 |
| Recall | 0.9920 | 0.9984 | 0.8538 | 0.9096 | 0.8232 | 0.7697 | 0.4313 | 0.0 |
| F-Measure | 0.9381 | 0.9072 | 0.8691 | 0.9003 | 0.8878 | 0.8402 | 0.5864 | 0.0 |

**Table 6.** Spark – confusion matrix – experiment 1.

| A | B | C | D | E | F | G | H | Class |
|---|---|---|---|---|---|---|---|---|
| 15199.0 | 24.0 | 83.0 | 0.0 | 12.0 | 3.0 | 0.0 | 0.0 | A - Anticipation |
| 13.0 | 11675.0 | 2.0 | 3.0 | 0.0 | 0.0 | 0.0 | 0.0 | B - Sadness |
| 657.0 | 234.0 | 6633.0 | 137.0 | 9.0 | 94.0 | 4.0 | 0.0 | C - Joy |
| 409.0 | 172.0 | 25.0 | 6262.0 | 12.0 | 3.0 | 1.0 | 0.0 | D - Trust |
| 41.0 | 436.0 | 120.0 | 25.0 | 2898.0 | 0.0 | 0.0 | 0.0 | E - Disgust |
| 79.0 | 580.0 | 3.0 | 48.0 | 8.0 | 2401.0 | 0.0 | 0.0 | F - Anger |
| 140.0 | 630.0 | 127.0 | 301.0 | 24.0 | 75.0 | 984.0 | 0.0 | G - Fear |

**Table 7.** Average runtime – spark single node and spark cluster.

| Experiment | Number of instances | Spark single node runtime (secs) | Spark 6 node cluster runtime (secs) |
|---|---|---|---|
| 1 | 174688 | 256.75 | 207.70 |

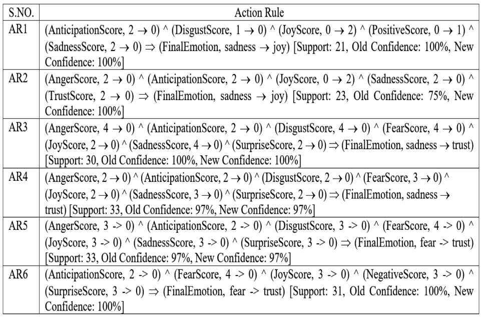
## 4.2 Action Rules

In this experiment, we extract action rules to identify what changes in attributes lead to change in emotion to a more positive state. For example, change from ‘sadness’ to ‘trust’; ‘sadness’ to ‘joy’. The dataset consists of continuous attributes which are discretized into intervals. The intervals are determined with the help of WEKA data mining software using unsupervised attribute discretization [32]. The following are the list of parameters set to discretize the data, 174688 instances, Weka – unsupervised discretize filter with 5 bins and equal frequency binning. We use the following attributes AngerScore, TrustScore, FearScore, SadnessScore, AnticipationScore, DisgustScore, SurpriseScore, JoyScore, PositiveScore, NegativeScore, LoveScore, PeopleScore, MessageScore, UserFollowersCount, UserFavoritesCount, UserFriendsCount, Tweet- Source, FinalEmotion from the original dataset. Discretization for the numeric attributes are shown in Table 9. The dataset with 174688 instances is divided into 100 parts based on the target class attribute ‘FinalEmotion’. Action rules are generated for one part of the dataset with 1439 instances and 18 attributes listed above. The Table 8 gives list of parameters used for action rule generation.

Figure 6 shows sample action rules generated. Let us consider the action rule AR1, this rule suggest possible changes to achieve a desirable emotional state of ‘joy’. The action rule is interpreted as follows: If the user tends to use more positive words as

**Table 8.** Action rule - parameters.

| Parameter | Values |
|---|---|
| Stable attributes | LoveScore, PeopleScore, MessageScore |
| Decision attribute | FinalEmotion |
| Support | 20 |
| Confidence | 30 |



| S.NO. | Action Rule |
|---|---|
| AR1 | (AnticipationScore, 2 → 0) ^ (DisgustScore, 1 → 0) ^ (JoyScore, 0 → 2) ^ (PositiveScore, 0 → 1) ^ (SadnessScore, 2 → 0) ⇒ (FinalEmotion, sadness → joy) [Support: 21, Old Confidence: 100%, New Confidence: 100%] |
| AR2 | (AngerScore, 2 → 0) ^ (AnticipationScore, 2 → 0) ^ (JoyScore, 0 → 2) ^ (SadnessScore, 2 → 0) ^ (TrustScore, 2 → 0) ⇒ (FinalEmotion, sadness → joy) [Support: 23, Old Confidence: 75%, New Confidence: 100%] |
| AR3 | (AngerScore, 4 → 0) ^ (AnticipationScore, 2 → 0) ^ (DisgustScore, 4 → 0) ^ (FearScore, 4 → 0) ^ (JoyScore, 2 → 0) ^ (SadnessScore, 4 → 0) ^ (SurpriseScore, 2 → 0) ⇒ (FinalEmotion, sadness → trust) [Support: 30, Old Confidence: 100%, New Confidence: 100%] |
| AR4 | (AngerScore, 2 → 0) ^ (AnticipationScore, 2 → 0) ^ (DisgustScore, 2 → 0) ^ (FearScore, 3 → 0) ^ (JoyScore, 2 → 0) ^ (SadnessScore, 3 → 0) ^ (SurpriseScore, 2 → 0) ⇒ (FinalEmotion, sadness → trust) [Support: 33, Old Confidence: 97%, New Confidence: 97%] |
| AR5 | (AngerScore, 3 -> 0) ^ (AnticipationScore, 2 -> 0) ^ (DisgustScore, 3 -> 0) ^ (FearScore, 4 -> 0) ^ (JoyScore, 3 -> 0) ^ (SadnessScore, 3 -> 0) ^ (SurpriseScore, 3 -> 0) ⇒ (FinalEmotion, fear -> trust) [Support: 33, Old Confidence: 97%, New Confidence: 97%] |
| AR6 | (AnticipationScore, 2 -> 0) ^ (FearScore, 4 -> 0) ^ (JoyScore, 3 -> 0) ^ (NegativeScore, 3 -> 0) ^ (SurpriseScore, 3 -> 0) ⇒ (FinalEmotion, fear -> trust) [Support: 31, Old Confidence: 100%, New Confidence: 100%] |

**Fig. 6.** Sample action rules.

**Table 9.** Discretization parameters.

| Attribute | Bins | ValueSet |
|---|---|---|
| AngerScore | -infinity, 0.002068, 0.997299, 1.007317, 2.0893, infinity | 0,1,2,3,4 |
| TrustScore | -infinity, 0.011484, 0.935696, 1.01071, 2.01071, infinity | 0,1,2,3,4 |
| FearScore | -infinity, 0.003022, 0.990587, 1.00374, 2.06263, infinity | 0,1,2,3,4 |
| SadnessScore | -infinity, 0.004326, 0.973808, 1.00306, 2.00306, infinity | 0,1,2,3,4 |
| AnticipationScore | -infinity, 0.324121, 0.992358, 1.00685, 2.00551, infinity | 0,1,2,3,4 |
| DisgustScore | -infinity, 0.000009, 0.997325, 1.00053, 2.00085, infinity | 0,1,2,3,4 |
| SurpriseScore | -infinity, 0.000056, 0.999872, 1.00141, 2.00545, infinity | 0,1,2,3,4 |
| JoyScore | -infinity, 0.001784, 0.999909, 1.00515, 2.00515, infinity | 0,1,2,3,4 |
| PositiveScore | -infinity, 0.5, 1.5, 2.5, 3.5,infinity | 0,1,2,3,4 |
| NegativeScore | -infinity, 0.5, 1.5, 2.5, 3.5,infinity | 0,1,2,3,4 |
| UserFollowersCount | -infinity, 105.5, 307.5, 656.5, 1662.5,infinity | 0,1,2,3,4 |
| UserFavoritesCount | -infinity, 575.5, 2570.5, 7123.5, 19418.5,infinity | 0,1,2,3,4 |
| UserFriendsCount | -infinity,146.5, 310.5, 574.5, 1253.5, infinity | 0,1,2,3,4 |

denoted by (JoyScore, 0 → 2) and (PositiveScore, 0 → 1), and reduce the words related to negative emotions like disgust, sadness and anticipation as denoted by (DisgustScore, 1 → 0) and (SadnessScore, 2 → 0) and (AnticipationScore, 2 → 0), then it is possible to change the emotion from 'sadness' to 'joy'. In that case, the emotion associated with this user tweet can be classified as 'joy', and we expect that the user is feeling more positive.

## 5 Conclusion

In this work, we automatically detect user emotion from tweet data using the NRC Emotion Lexicon [21, 22] to label the Emotion class for our data. We use Support Vector Machine with Multiclass classification in particular ONE-AGAINST-ALL implementation in both WEKA data mining software [32] and Apache Spark system

[31] over Hadoop 6 node Cluster for big data scalability. We achieve accuracy of 84.9% to 88.01%. The Spark system is able to scale to BigData with six node cluster, as the data is partitioned into several sets and processed in parallel at each cluster node. This is an extension of previous study [33] of finding user emotions from tweet data using the NRC emotion lexicon to label the emotion class for our data. In the previous work, we examined several classifiers including Decision Tree, Decision Forest, and Decision Table Majority. In this work, we extract Action Rules to identify what factors can be improved in order for a user to attain a more desirable positive emotion. We suggest actions that can be undertaken to reclassify user emotion from a negative emotion to more positive emotion. For instance from 'sadness' to 'joy', 'sadness' to 'trust', and 'fear' to 'trust'. In the future, we plan further test with larger social networking data. We also plan to apply this system for customer surveys and education evaluations.

## References

1. M.-W. Dictionary, "Merriam-webster" (2002). http://www.mw.com/home.htm
2. Honey, C., Herring, S.C.: Beyond microblogging: conversation and collaboration via twitter. In 42nd Hawaii International Conference on System Sciences, 2009. HICSS 2009, pp. 1–10. IEEE (2009)
3. Chang, Y., Tang, L., Inagaki, Y., Liu, Y.: What is tumblr: a statistical overview and comparison. ACM SIGKDD Explor. Newsl. **16**(1), 21–29 (2014)
4. Sullivan, D.: Comscore media metrix search engine ratings. Search Engine Watch **21** (2006)
5. Java, A., Song, X., Finin, T., Tseng, B.: Why we twitter: understanding microblogging usage and communities. In: Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 Workshop on Web Mining and Social Network Analysis, pp. 56– 65. ACM (2007)
6. Chang, H.-C.: A new perspective on twitter hashtag use: diffusion of innovation theory. Proc. Assoc. Inf. Sci. Technol. **47**(1), 1–4 (2010)
7. Hasan, M., Agu, E., Rundensteiner, E.: Using hashtags as labels for supervised learning of emotions in twitter messages. In: ACM SIGKDD Workshop on Health Informatics. New York, USA (2014)
8. Gupta, N., Gilbert, M., Fabbrizio, G.D.: Emotion detection in email customer care. Comput. Intell. **29**(3), 489–505 (2013)
9. D'Alfonso, S., Santesteban-Echarri, O., Rice, S., Wadley, G., Lederman, R., Miles, C., Gleeson, J., Alvarez-Jimenez, M.: Artificial intelligence-assisted online social therapy for youth mental health. Front. Psychol. **8**, 796 (2017)
10. Tantam, D.: The machine as psychotherapist: impersonal communication with a machine. Adv. Psychiatr. Treat. **12**(6), 416–426 (2006)
11. Kaur, H.: Actionable rules: issues and new directions. In: World Enformatika Conference - WEC (5), pp. 61–64. Citeseer (2005)
12. He, Z., Xu, X., Deng, S., Ma, R.: Mining action rules from scratch. Expert Syst. Appl. **29**(3), 691–699 (2005)
13. Mishne, G., et al.: Experiments with mood classification in blog posts. In: Proceedings of ACM SIGIR 2005 Workshop on Stylistic Analysis of Text for Information Access, vol. 19, pp. 321–327 (2005)

14. Danisman, T., Alpkocak, A.: Feeler: emotion classification of text using vector space model. In: AISB 2008 Convention Communication, Interaction and Social Intelligence, vol. 1, p. 53 (2008)
15. Mohammad, S.M.: #emotional tweets. In: Proceedings of the First Joint Conference on Lexical and Computational Semantics - Volume 1: Proceedings of the Main Conference and the Shared Task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation, pp. 246–255. Association for Computational Linguistics, Stroudsburg (2012)
16. Roberts, K., Roach, M.A., Johnson, J., Guthrie, J., Harabagiu, S.M.: Empatweet: annotating and detecting emotions on twitter. In: LREC, vol. 12, pp. 3806–3813. Citeseer (2012)
17. Ekman, P.: An argument for basic emotions. Cogn. Emot. **6**(3–4), 169–200 (1992)
18. Purver, M., Battersby, S.: Experimenting with distant supervision for emotion classification. In: Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics, pp. 482–491. Association for Computational Linguistics (2012)
19. Ranganathan, J., Irudayaraj, A.S., Tzacheva, A.A.: Action rules for sentiment analysis on twitter data using spark. In: 2017 IEEE International Conference on Data Mining Workshops (ICDMW), pp. 51–60, November 2017
20. Makice, K.: Twitter API: Up and Running: Learn How to Build Applications with the Twitter API. O'Reilly Media, Inc., Beijing (2009)
21. Mohammad, S.M., Turney, P.D.: Crowdsourcing a word– emotion association lexicon. Comput. Intell. **29**(3), 436–465 (2013)
22. Mohammad, S.M., Turney, P.D.: 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)
23. Mohammad, S.M., Kiritchenko, S.: Using hashtags to capture fine emotion categories from tweets. Comput. Intell. **31**(2), 301–326 (2015)
24. Chang, C.-C., Lin, C.-J.: Libsvm: a library for support vector machines. ACM Trans. Intell. Syst. Technol. (TIST) **2**(3), 27 (2011)
25. Hsu, C.-W., Chang, C.-C., Lin, C.-J., et al.: A practical guide to support vector classification (2003)
26. Gunn, S.R., et al.: Support vector machines for classification and regression. ISIS Tech. Rep. **14**(1), 5–16 (1998)
27. Hsu, C.-W., Lin, C.-J.: A comparison of methods for multiclass support vector machines. IEEE Trans. Neural Networks **13**(2), 415–425 (2002)
28. Grzymala-Busse, J.W.: A new version of the rule induction system lers. Fundamenta Informaticae **31**(1), 27–39 (1997)
29. Dardzinska, A., Ras, Z.W.: Extracting rules from incomplete decision systems: system ERID. In: Lin, T.Y., Ohsuga, S., Liau, C.J., Hu, X. (eds.) Foundations and Novel Approaches in Data Mining. Studies in Computational Intelligence, vol. 9, pp. 143–153. Springer (2005)
30. Zaharia, M., Xin, R.S., Wendell, P., Das, T., Armbrust, M., Dave, A., Meng, X., Rosen, J., Venkataraman, S., Franklin, M.J., et al.: Apache spark: a unified engine for big data processing. Commun. ACM **59**(11), 56–65 (2016)
31. Meng, X., Bradley, J., Yavuz, B., Sparks, E., Venkataraman, S., Liu, D., Freeman, J., Tsai, D., Amde, M., Owen, S., et al.: Mllib: machine learning in apache spark. J. Mach. Learn. Res. **17**(1), 1235–1241 (2016)
32. Witten, I.H., Frank, E., Hall, M.A., Pal, C.J.: Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann, San Francisco (2016)

33. Ranganathan, J., Hedge, N., Irudayaraj, A., Tzacheva, A.: Automatic detection of emotions in twitter data - a scalable decision tree classification method. In: Proceedings of the RevOpID 2018 Workshop on Opinion Mining, Summarization and Diversification in 29th ACM Conference on Hypertext and Social Media (2018)
34. Tzacheva, A.A., Sankar, C.C., Ramachandran, S., Shankar, R.A.: Support confidence and utility of action rules triggered by meta-actions. In: 2016 IEEE International Conference on Knowledge Engineering and Applications (ICKEA), pp. 113–120. Singapore (2016). https://doi.org/10.1109/ickea.2016.7803003