International Journal of Data Science and Analytics (2019) 7:35–51
https://doi.org/10.1007/s41060-018-0096-z

REGULAR PAPER

Automatic emotion detection in text streams by analyzing Twitter data

Maryam Hasan[1] · Elke Rundensteiner[1] · Emmanuel Agu[1]

Received: 14 March 2017 / Accepted: 16 January 2018 / Published online: 9 February 2018

Abstract

Techniques to detect the emotions expressed in microblogs and social media posts have a wide range of applications including, detecting psychological disorders such as anxiety or depression in individuals or measuring the public mood of a community. A major challenge for automated emotion detection is that emotions are subjective concepts with fuzzy boundaries and with variations in expression and perception. To address this issue, a dimensional model of affect is utilized to define emotion classes. Further, a soft classification approach is proposed to measure the probability of assigning a message to each emotion class. We develop and evaluate a supervised learning system to automatically classify emotion in text stream messages. Our approach includes two main tasks: an offline training task and an online classification task. The first task creates models to classify emotion in text messages. For the second task, we develop a two-stage framework called EmotexStream to classify live streams of text messages for the real-time emotion tracking. Moreover, we propose an online method to measure public emotion and detect emotion burst moments in live text streams.

Keywords Supervised emotion learning · Real-time emotion detection · Twitter events analysis · Public emotion sensing · Text stream classification · Soft classification

1 Introduction

Emotion plays a critical role in our daily performance affecting many aspects of our lives including social interaction, behavior, attitude, and decision-making [1]. Understanding human emotion patterns and how the people feel plays an essential role in various applications such as public health and safety, emergency response, and urban planning.

Text is a particularly important source of data for detecting emotion because the bulk of textual data ranging from microblogs, emails, to SMS messages on a smart phone that has become increasingly available. The rapid growth of emotion-rich textual data makes a necessity to automate identification and analysis of people's emotion expressed in text [1].

1.1 Emotion in social networks

Social networks and microblogging tools (e.g., Twitter, Facebook) are increasingly used by individuals to share their opinions and feelings in the form of short text messages (e.g., texts about normal life and opinion on current issues and events) [2]. These messages (commonly known as tweets or microblogs) may also contain indicators of emotions of individuals such as happiness, anxiety, and depression. In fact, social networks contain a large corpus of public real-time data that is rich with emotional content. This makes them appropriate data sources for behavioral studies, especially for studying emotions of individuals as well as larger populations. Therefore, social networks such as Twitter provide valuable information to observe crowd emotion and behavior and study a variety of human behavior and characteristics [3].

Increasing evidence suggests that emotion detection and screening built around social media [4–7] will be effective in many applications. In particular, Twitter provides valuable opportunities to observe public mood and behavior. The development of robust textual emotion sensing technologies promises to have a substantial impact on public and individual health and urban planning. Such emotion mining tools, once available, could potentially be employed in a large variety of applications ranging from population level studies of

✉ Maryam Hasan
mhasan@wpi.edu; mhasan@cs.wpi.edu

Elke Rundensteiner
rundenst@cs.wpi.edu

Emmanuel Agu
emmanuel@cs.wpi.edu

[1] Computer Science Department, Worcester Polytechnic Institute, Worcester, MA, USA

emotions, the provision of mental health counseling services over social media, and other emotion management applications. The census bureau and other polling organizations may be able to use the emotion mining technology to estimate the percentage of people in a community experiencing certain emotions and correlate this with current events and various other aspects of urban living conditions. This type of technology can also enhance early outbreak warning for public health authorities so that a rapid action can take place [8].

Moreover, the emotion mining tools could also be used by counseling agencies to monitor emotional states of individuals or to recognize anxiety or systemic stressors of populations [9]. For instance, university counseling centers could be warned early about distressed students that may require further personal assessment.

1.2 Challenges of detecting emotion in social networks

Our goal is to detect emotion in social networks by classifying text messages into several classes of emotion. To achieve this goal, the major challenges discussed below must be tackled:

- *Casual style of microblog data* Text messages are usually written in a casual style. They may contain numerous grammatical and spelling errors along with slang words. While the use of informal language and short messages has been previously studied in the context of sentiment analysis [10–13], the use of such language in the context of emotion mining has been much less studied.
- *Semantic ambiguity of text messages* Human emotions as well as the texts expressing them are ambiguous and subjective. This makes it difficult to accurately infer and interpret the author's emotional states.
- *Fuzzy boundaries of emotion classes* Emotions are complex concepts with fuzzy boundaries and with variations in expression. Thus, modeling and analyzing the human affective behavior is a challenge for automated systems [14].
- *Difficulty of emotion annotation* In order to train an automatic classifier, supervised learning methods require labeled data. It would be time-consuming, tedious and labor-intensive to manually label text messages for the purpose of training a classifier for emotion detection.
- *Numerous topics and emotional states* The large breadth of topics discussed on social networks makes it challenging to manually create a comprehensive corpus of labeled data that covers all possible emotional states.
- *Inconsistent annotators* While crowdsourcing emotion labels have been explored, human annotators may not be reliable. A human annotator's judgement of the emotions in a text message is likely to be subjective and inconsistent. Consequently, different annotators may classify the same text message into different emotion classes, as confirmed by our user study in Sect. 5.

1.3 Proposed approach to detect emotion in text stream messages

To detect and analyze the emotion expressed in text messages, we develop a supervised machine learning approach to automatically classify the messages into their emotional states. Our approach includes two main tasks: an offline training task and an online classification task. During the first task, we collect a large dataset of emotion-labeled messages from Twitter. The messages are preprocessed and used to train emotion classification models. The second task utilizes the created models to classify live streams of tweets for real-time emotion tracking in a geographic location (e.g., a city). For the second task we develop a two-stage framework called EmotexStream. A binary classifier is created in the first stage to separate tweets with explicit emotion from tweets without emotion. The second stage utilizes our emotion classification models for a fine-grained (i.e., multi-class) emotion classification of tweets with explicit emotion.

While supervised learning methods achieve high accuracy, they require a large corpus of texts labeled with the emotion classes they express [15]. Prior works have mostly utilized manually labeled data. Crowdsourcing is a popular approach for labeling data, in which humans manually infer and then annotate each message with the emotion it expresses [2,4,5]. Crowdsourcing tools such as Amazon's mechanical turk facilitate access to manual data labelers. However, manually labeling of Twitter messages with the emotions they express faces numerous challenges as previously outlined, including the inconsistency of human labelers (See Sect. 1.2). Therefore, instead we investigate using hashtags (user-selected keywords) in Twitter messages as viable alternative to manual labeling. The use of hashtags in tweets is very common. Twitter contains millions of different user-defined hashtags. Wang et al. showed that 14.6% of tweets in a sample of 0.6 million tweets had at least one hashtag [15]. We make the observation that in many cases the hashtag keywords may correspond to the author's own classification of the main topics of their message. A study by Wang et al. showed that emotion hashtags in about 93% of their sample tweets are relevant and reflect the writer's emotion [1].

We thus conjecture that emotional hashtags inserted by authors indicate the main emotion expressed by their Twitter message. For example, a tweet with the hashtag "#depressed" can be interpreted as expressing a depressed emotion, while a tweet containing the hashtag "#excited" as expressing excitement. By using embedded hashtags to automatically label the emotions expressed in text messages, we build a large corpus of labeled messages to train classifiers with no manual effort. This approach overcomes the need for manual labeling and yields a completely automatic scheme for labeling a massive repository of Twitter messages. This strategy could equally be applied in other mining applications where labeling is required.

Another challenge for automated emotion detection is that emotions are complex concepts with fuzzy boundaries and with individual variations in expression and perception. We address this issue using a two-pronged approach. First, we define the emotion classes based on the Circumplex model of affect [16]. Instead of a small number of discrete categories, this model defines the emotion in terms of latent dimensions (e.g., arousal and valence). Second, a soft (i.e., fuzzy) classification approach is proposed, which classifies each message into multiple emotion classes with different probabilities (i.e., weights), instead of forcing each message to be in one emotion class only.

This paper is an extension of our preliminary results published in [9,17]. We first studied supervised learning to classify emotion in texts, and the idea of considering Twitter hashtags as automatic emotion labels [17]. We then validated the effectiveness of utilizing our hashtag-based labeling concept through two user studies, one with psychology experts and the other with the general crowd [9]. In this journal paper, we now extend our previous system and develop a two-stage framework, called EmotexStream that performs online emotion analysis on live streams of text messages. The first stage of EmotexStream separates tweets with explicit emotion from tweets without any emotion using a binary classifier. The second stage classifies the tweets with explicit emotion into fine-grained emotion classes. Furthermore, we deploy EmotexStream to measure public emotion and investigate its temporal distribution during major public events in a geographic location (e.g., a city). For this, we develop an online method to detect emotion bursts in live stream of messages. In particular, this paper makes the following major contributions:

- We develop the Emotex system to automatically classify emotion expressed in text messages. We evaluate the classification accuracy of Emotex by comparing it with the accuracy of the lexical approach.
- We utilize a soft (fuzzy) classification approach to measure the probability of assigning a message into each emotion class, in addition to a typical classification that simply assigns one single emotion class to each text message in a deterministic manner.
- We run ample experiments using offline data to train classification models and evaluate Emotex system and report its soft and hard classification results.
- We develop a two-stage framework called EmotexSream to classify live streams of tweets in the wild. The first stage separates tweets with explicit emotion from tweets without any emotion using a binary classifier. The second stage deploys Emotex to conduct a fine-grained emotion classification on tweets with explicit emotion.
- We evaluate EmotexStream framework by running some experiments using live and unfiltered streams of tweets in the wild.
- We propose an online method to measure public emotion and detect emotion-intensive moments, which can be used for real-time emotion tracking.

The rest of the paper is organized as follows. Section 2 describes different models of emotion. Details of our proposed methods to detect and analyze emotion in text streams are illustrated in Sect. 3. Section 4 includes our extensive experimental results about different tasks of our approach. Evaluating our labeling method is described in Sect. 5. Section 6 includes related work on emotion detection in text. Finally, we conclude our paper in Sect. 7.

2 Models of emotion

The emotion models have mainly been studied based on two fundamental approaches: basic emotions model and dimensional model [18].

2.1 Basic emotions model

According to the *basic emotion model* humans have a small set of basic emotions, which are discrete and detectable by an individual's verbal/nonverbal expression [19]. Researchers have attempted to identify a number of basic emotions which are universal among all people and differ one from another in important ways. A popular example is a cross-cultural study by Ekman et al. [19], in which they concluded that six basic emotions are anger, disgust, fear, happiness, sadness, and surprise. Subsequently, many works in the field of emotion detection in texts have been conducted based on this basic emotion model [20–23]. For example, Bollen et al. extracted six dimensions of affect including tension, depression, anger, vigor, fatigue, confusion from Twitter to model public emotion [20].

However, the main drawback of such basic emotion models is that there is no consensus among theorists on which human emotions should be included in the basic set of emotions. Moreover, the basic emotions doesn't cover all the variety of emotion expressed by humans in texts. People usually express nonbasic, subtle and complex emotions. This problem can't be resolved by using a finer granularity, because the emotions expressed in texts are ambiguous and subjective. For instance, "surprise" as a basic emotion can indicate negative, neutral or positive valence. Also using a finer granularity of emotion makes the distinction of one emotion from another an issue in emotion classification.

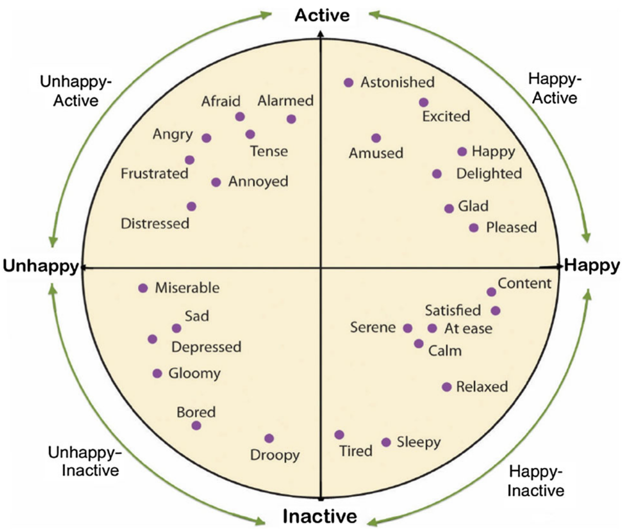

Fig. 1 Circumplex model of affect including 28 affect words by [16]

Therefore, a small number of discrete emotions may not reflect the complexity of the affective states conveyed by humans [18].

2.2 Dimensional model of emotion

In contrast to the basic emotion model which defines discrete emotions, the *dimensional model* defines emotion on a continuous scale. This model characterizes human emotions by defining their positions along two or three dimensions. Many dimensional models incorporate two fundamental dimensions of emotions namely, valence (i.e., pleasure) and arousal (i.e., activation or stimulation) [18].

The most widely used dimensional model is the Circumplex model of affect proposed by Russell [16]. As shown in Fig. 1, the model suggests that emotions are distributed in a two-dimensional circular space, containing valence and arousal dimensions. The horizontal axis presents pleasure and measures how positive or negative a person feels. The vertical axis presents activation and measures if one is likely to take an action.

Although the Circumplex model is a well-known model and has long been validated and studied by emotion and cognition theorists, it has rarely been used by computational approaches for automatic emotion analysis in texts [24]. In our emotion classification work, we utilize the Circumplex model by considering four major classes of emotion: Happy-Active, Happy-Inactive, Unhappy-Active, and Unhappy-Inactive. As shown in Fig. 1, the defined four classes of emotion are distinct, yet describe a wide range of emotional states as they cover four dimensions of the Circumplex model.

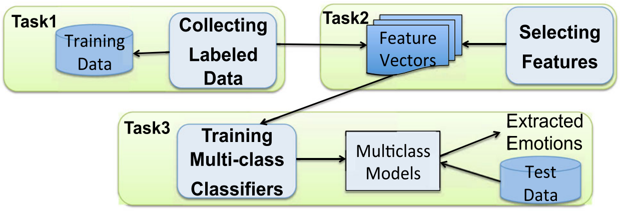

Fig. 2 Model of emotex

3 Proposed approach to detect emotion in text stream messages

To detect and analyze the emotion expressed in text stream messages we develop a supervised machine learning approach to automatically classify the messages into their emotional states. Our approach includes two main tasks: an offline training task and an online classification task. The first task develops a system called Emotex to create models for classifying emotion. Emotex collects a large dataset of emotion-labeled messages from Twitter. The messages are then preprocessed and converted into the feature vectors to train emotion classification models. The second task utilizes the created models to classify live streams of tweets for real-time emotion tracking. For this task, we develop a two-stage framework called EmotexStream. EmotexStream creates a binary classifier to separate tweets with explicit emotion from tweets without emotion. Then it utilizes our emotion classification models for a fine-grained emotion classification of tweets with explicit emotion.

Furthermore, we develop an online method to measure public emotion and detect emotion bursts in live streams of tweets posted in a geographic location. Details of the proposed tasks and methods are given in the following.

3.1 Emotex: a supervised learning model to classify emotion in text messages

We develop a supervised learning system called Emotex to classify texts into our defined classes of emotion described in Sect. 2.2. Emotex is developed as an offline system and includes three parts. The first part involves data acquisition and collecting training data. The second part is related to feature selection and the third part creates the emotion classifiers. Figure 2 shows the process flow of Emotex. First, we collect Twitter messages and annotate them with emotion labels to develop a dataset to train classification models. Second, we select certain features and convert each tweet in the training set into a feature vector. We then utilize the feature vectors annotated with emotion labels to train classifiers. The result is a model that can classify unlabeled messages into an appropriate emotion class. This section now describes each part of the Emotex pipeline.

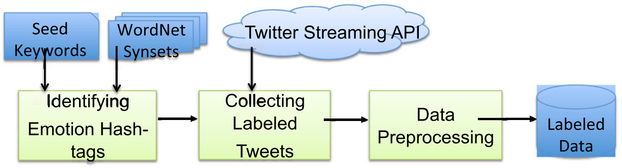

Fig. 3 Model of labeled data collection

3.1.1 Collecting labeled data

We utilize hashtags to automatically annotate text messages with emotion and build a large corpus of emotion-labeled messages. These messages then serve as a labeled dataset for training classifiers. Figure 3 shows the steps of collecting labeled data. We first need to define a list of emotion hashtags to collect emotion-labeled messages. For this, we exploit the set of 28 affect words from the Circumplex model (as shown in Fig. 1) as the initial set of keywords and extend them using WordNet's synsets [25]. We use the extended set of keywords to detect emotion hashtags. Then, we collect tweets which contain one or more hashtags that fall in our defined list of emotion hashtags. This way we assure that we have tweets labeled with our defined emotion classes described in Sect. 2.2. Hashtags that are directly interleaved in the actual tweet text are more likely to represent a part of the content of the tweet itself [1,2]. Therefore, we only collect the tweets which contain the emotion hashtags at the end. We also don't collect retweets, which begin with the"RT" keyword.

Using this approach we are able to collect a large number of tweets with various emotion hashtags with no manual effort. Another major advantage of this approach is that it gives us direct access to the author's own intended emotional state, instead of relying on the possibly inconsistent and inaccurate interpretations of third-party annotators about what an author of a tweet may have felt. We utilize Twitter's stream API to automatically collect tweets and filter them by emotion hashtags. After collecting the same number of tweets for each emotion class, the labeled tweets are then preprocessed to mitigate misspellings and casual language used in Twitter using the following rules:

- *User IDs and URLs* In addition to the message body, tweets contain the ID of the user and URL links. They are marked separately for later processing.
- *Text normalization* Tweets often contain abbreviations and informal expressions. All abbreviations are expanded (e.g., "won't" to "will not"). Words with repeated letters are common. Any letter occurring more than two times consecutively is replaced with one occurrence. For instance, the word "happyyyy" will be changed into "happy".
- *Conflicting hashtags* Some tweets may contain hashtags from different emotion classes. For example tweet "Got a job interview with At&t…#nervous #happy.", includes the hashtag #nervous from Unhappy-Active class and the tag #happy from Happy-Active class. Tweets with conflicting hashtags are removed from our labeled data, as they illustrate a mixture of different emotions.
- *Hashtags at end of tweets* We consider emotion hashtags at the end of the tweets as emotion labels. Therefore, as part of preprocessing, emotion hashtags are stripped off from the end of tweets. For instance, the tags "#disappointed" and "#sad" are removed from the tweet "No one wants to turn up today. #disappointed #sad". Hashtags that are directly interleaved in the actual tweet text represent part of the content of the tweet and are not removed.

3.1.2 Feature selection for capturing emotion

In order to train a classifier from labeled data, we represent each tweet as a vector of numerical features. Thus, a set of features that illustrate the emotion expressed by each tweet is needed. Feature selection plays an important role in emotion classification. We investigate the effectiveness of different features. We use single words, also known as unigrams, as the baseline features for comparison. Other features explored include emoticons, punctuations, and negations.

Unigram Features: Unigrams or single word features have been widely used to capture sentiment or emotion in text [10,11,21]. Let $\{f_1, f_2, \ldots, f_m\}$ be our predefined set of unigrams that can appear in a tweet. Each feature f_i in this vector is a word occurring in the list of tweets in our dataset. However, with the large breadth of topics discussed in microblogs, the number of words in our input dataset tends to be extremely large. Thus, the feature vector of each message would become excessively large and sparse (i.e., most features will have a value of zero). To overcome the problem of this high-dimensional feature space, we select an emotion lexicon as the set of unigram features. As a result, our feature space only contains the emotional words from the emotion lexicons instead of all the words in our training dataset. This method reduces the size of feature space dramatically, with minimal loss of informative terms.

We use different emotion lexicons in our system, including ANEW lexicon (Affective Norms for English Words) [26], LIWC dictionary (Linguistic Inquiry and Word Count) [27], and AFINN [28]. LIWC is a dictionary of several thousands words and prefixes, grouped into psychological categories. We use emotion-indicative categories including positive emotions, negative emotions, anxiety, anger, sadness, and negations. ANEW lexicon contains 2477 affect words, each rated for its valence and arousal on a 1–9 scale. AFINN was created to include a new word list specifically for microblogs.

Emoticon features Other than unigrams, emoticons are also likely to be useful features to classify emotion in texts as they are textual portrayals of emotion in the form of icons. Emoticons tend to be widely used in sentiment analysis. Go et al. and Pak et al. [10,11] used the western-style emoticons to collect labeled data. There are many emoticons to express happy, sad, angry or sleepy emotion. The list of emoticons that we use can be found in our paper [17].

Punctuation features Other features potentially helpful for emotion detection are punctuations (i.e., question mark, exclamation mark and combination of them). Users often use exclamation marks when they want to express their strong feelings. For instance, the tweet "I lost 4lb in 3 days!!" expresses strong happiness and the tweet "we're in december, which means one month until EXAMS!!!" represents a high level of stress. The exclamation mark is sometimes used in conjunction with the question mark, which in many cases appears to convey a sense of astonishment. For example the tweet "You don't even offer high speed, yet you keep overcharging me?!" indicates an astonished and annoyed feeling.

Negation features As our last feature, we select negation to address errors caused by tweets that contain negated phrases like "not sad" or "not happy". For example the tweet, "I'm not happy about this trade." should not be classified as a happy tweet, even though it has a happy unigram. To tackle this problem we define negation as a separate feature. We select the list of phrases indicating negation from the LIWC dictionary.

3.1.3 Classifier selection for emotion detection

A number of classification methods have been applied for text categorization, including Bayesian classifiers, decision trees, nearest neighbor classifiers, and support vector machines (SVM). To classify emotion we explored three different classifiers. We selected Naive Bayes as a probabilistic classifier, SVM as a decision boundary classifier, and decision tree as a rule based classifier.

One of the challenges of automated emotion detection is that emotions are complex concepts with fuzzy boundaries and with many variations in expression. Also, emotion perception is naturally subjective. Thus, it is difficult to achieve a consensus to which emotion class each text message belongs to. As shown in our user studies described in Sect. 5, people often have different perceptions about emotion expressed in texts. Furthermore, a small number of discrete emotion classes may not reflect the complexity of the emotional states conveyed by humans. Typical classifiers assume clearly demarcated and nonoverlapping classes. They may not assign emotion labels to some messages with high confidence and classify them either incorrectly or correctly with low confidence. Therefore, simply assigning one single emotion class to each text message in a deterministic manner may not perform well in practice.

To overcome this issue, we use a two-pronged approach. First, we define the emotion classes based on a dimensional model (See Sect. 2.2). Second, a soft (fuzzy) classification approach is proposed to measure the strength of each emotion class in association with the message under classification. In soft classification, the prediction results become less explicit by assigning each message a soft label that indicates how likely each emotion would be perceived. More details about hard and soft classification of emotion are described bellow.

Hard and Soft Classification of Emotion For classifying emotion, we utilize two types of classification: soft and hard classification. In general, a classifier is a function that assigns an emotion label y to an input feature vector x:

$$y = f(x), \quad x \in X, \quad y \in Y \tag{1}$$

where X is the set of all feature vectors from the tweets in the input dataset, and Y is the set of emotion labels.

Some classifiers such as support vector machines make decision boundaries between different classes. Other classifiers are probabilistic classifiers meaning that they assign a probability distribution over a set of classes to an input $x \in X$.

$$P(Y = y|x), \quad x \in X, \quad y \in Y \tag{2}$$

In hard classification, each message can only belong to one and only one class. Soft classifiers measure the degree to which a message belongs to each class, rather than dedicating the message to a specific class [29]. In decision boundary classifiers, soft labels can be estimated based on decision scores. In probabilistic classifiers, soft labels can refer to the class conditional probabilities, and a hard classification label can be produced based on the largest estimated probability.

$$y = \max_y \{P(Y = y|x), \quad x \in X, \quad y \in Y\} \tag{3}$$

For example, the tweet "I can live for months on a good compliment" is 65% likely to be happy, 18% likely to be relaxed, 9% likely to be angry, and 8% likely to be sad. Since the maximum probability of the tweet is 65%, it can be assigned to the happy class.

Naive Bayes and logistic regression are probabilistic classifiers which produce a probability distribution over output classes. Other models such as support vector machines do not produce probabilities. They instead return decision scores which are proportional to the distance from the separating hyperplane. They classify input data (here, tweets) with certain decision scores, which can be considered as soft labels. However, these scores may not correspond with class membership probabilities, since the distance from the separating hyperplane is not exactly proportional to the chances of class membership [30]. Some methods have been developed to convert the results of these classifiers into class membership probabilities. A common method is to apply Platt scaling [31], which learns the following sigmoid function defined by the parameters A and B on the decision scores $s(x)$:

$$P(Y = y|x) = \frac{1}{1 + e^{As(x)+B}} \tag{4}$$

Zadrozny and Elkan proposed another method by using isotonic regression when sufficient training data are available [30].

3.2 EmotexStream: a framework for classifying live streams of text messages

After developing Emotex, we now aim to deploy the trained model to analyze live streams of tweets. However analyzing text in real time is challenging due to the noise and fast-paced nature of tweets in the wild. For this, we develop a two-stage approach for classifying live streams of tweets.

Twitter messages cover a wide range of subjects. However, since our focus is on emotion detection, we are only interested in processing messages that contain emotions. For instance, the tweet "I have a wonderful roommate" conveys a happy emotion and is a good input to our system. In contrast, the tweet "It’s time for bed" cannot be identified as expressing any type of emotion neither happy nor sad. Therefore, we aim to identify such tweets without emotion and eliminate them in a fast preclassification step. In fact, we decompose the emotion detection task into two subtasks. We first detect tweets without any identifiable emotion using a binary classifier. Then we conduct a fine-grained emotion classification on tweets with explicit emotion.

Figure 4 shows our emotion analysis pipeline in classifying the general stream of tweets. As it shows, after cleaning and preprocessing of tweets we categorize tweets into two general classes, namely emotion-present and emotion-absent tweets. For binary classification of tweets we develop an unsupervised method that utilizes emotion lexicons. Our binary classifier assumes that tweets with no emotion are the ones without any emotional or affective words. Therefore, it classifies tweets containing at least one affective or emotional word as emotion-present tweets, and classifies tweets without any affective word as emotion-absent tweets. As we described in Sect. 3.1.2, different emotion lexicons are available, including ANEW lexicon, LIWC dictionary, and AFINN. We utilize all the affective words from these three lexicons and create a comprehensive affective lexicon for our binary classification task. After binary classification, emotion tweets will then go through the feature selection and multi-class emotion classifier generated by our Emotex technology to classify them based on our defined classes of emotion.

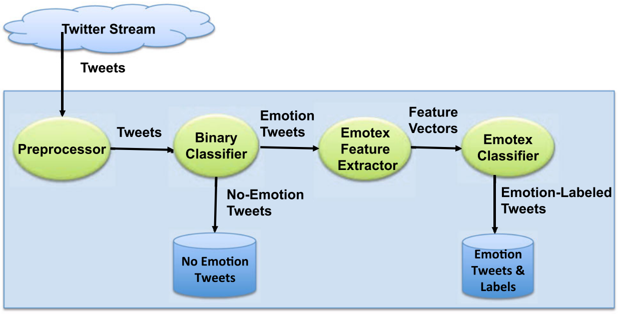

Fig. 4 EmotexStream: a two-stage approach to classify live streams of tweets

3.3 Proposed approach to detect emotion-intensive moments in live streams of messages

Detecting and measuring emotion in social networks such as Twitter enable us to observe crowd emotion and behavior. Using EmotexStream, we are able to classify live streams of tweets in real-time. We now aim to use our EmotexStream system to measure public emotion and detect emotion burst moments in live stream of tweets. We are looking for the percentage of people in a geographic location experiencing certain emotions during a specific time. The goal is to explore temporal distributions of aggregate emotion and detect temporal bursts in public emotion from live text streams. For this purpose, we first apply our EmotexStream system to automatically detect the emotion of people from their messages in live stream of tweets. As shown in Fig. 5, EmotexStream converts live text streams into streams of emotion classes. Then we aggregate the emotion stream of each class into a time-based histogram to analyze public emotion trends and discover emotion-evolving patterns over time. We propose an online method to measure public emotion and detect abrupt changes in emotion as emotion-intensive moments in live text streams [32]. Before describing our online method to detect important moments in social streams, we define some concepts in the context of tweet streams as below:

Definition 1 (*Emotion Stream*) An emotion stream S_E is a continuous sequence of time-ordered messages $M_1, M_2, \cdots M_r, \cdots$ from a tweet stream, such that each message M_i belongs to a specific emotion class $E_{c1} \in E_{\text{Class}}$ (E_{Class} is the set of predefined emotion classes defined in Sect. 2).

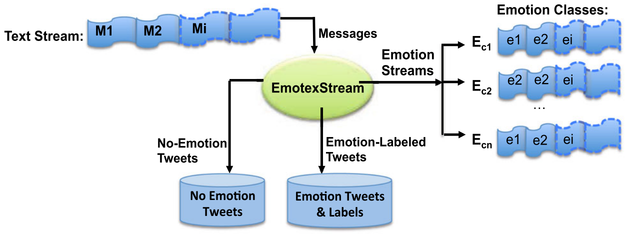

Fig. 5 Converting text streams into emotion streams using EmotexStream

In order to estimate the value of a specific emotion class E_{c1} among the people in a geographic location L during a time period $[T_1, T_2]$, we define a function as below:

$$E_{\text{public}}(T_1, T_2, L, E_{c1}) = \sum_{T_1 < T_i < T_2,\ L_i \in L} F(M_i, E_{c1}) \tag{5}$$

where $M_i =< U_i, T_i, L_i, C_i, E_i >$ is a tweet message in the emotion stream from the emotion class $E_i \in E_{\text{Class}}$, posted by user U_i in location $L_i \in L$, at the time $T_1 < T_i < T_2$, and $F(M_i, E_{c1})$ is an indicator function defined as below:

$$F(M_i, E_{c1}) = \begin{cases} 1 & \text{if } M_i \in E_{c1}, \\ 0 & \text{Otherwise.} \end{cases} \tag{6}$$

Using Eq. 5 we can quantify emotion of a population in a geographic location and during a time period. We can then analyze such emotion streams to detect temporal bursts of crowd emotion. These sudden bursts are characterized by a change in the fractional presence of messages in particular emotion classes. Formally, we define such abrupt changes as "emotion burst", which can point toward important moments. In order to detect emotion bursts, we determine the higher or the lower rate at which messages have arrived to an emotion class in the current time window of length W. Two parameters α and β are used to measure this evolution rate.

Definition 2 (*Emotion Burst*) An emotion burst over a temporal window of length W at the current time T_c is said to have occurred in a geographic region L, if the presence of a specific class emotion E_{c1} during a time period $(T_c - W, T_c)$ is less than the lower threshold α or greater than the upper threshold β.

In other words, we should have either

$$E_{\text{public}}(T_c - W, T_c, L, E_{c1}) \leq \alpha \tag{7}$$

or

$$E_{\text{public}}(T_c - W, T_c, L, E_{c1}) \geq \beta. \tag{8}$$

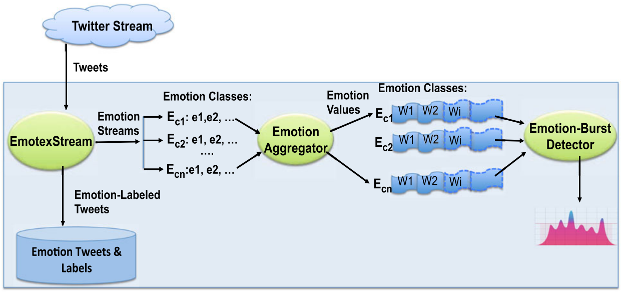

Fig. 6 Detecting emotion bursts in live text streams

Now we need to define the upper bound α and lower bound β of public emotion for each emotion class during a temporal window. If our algorithm is applied offline (i.e., all the tweets are available), the thresholds can be estimated from the average sum over the whole time period. However, in the online approach all the tweets are not available. Therefore, in the online approach, we compute the thresholds from the tweets in a temporal sliding window, where the size of the moving window is a parameter.

Figure 6 presents our system for detecting important moments in live text streams. Emotion streams can be created by applying EmotexStream system to classify tweets arriving in a stream. Let $e_1, \ldots e_i, \cdots e_n$ denote the emotion values of class E_{c1} of the tweets posted within a temporal window of length W in an emotion stream (n is the number of tweets posted within W). Apparently, $e_1, \ldots e_i, \cdots e_n$ are independent 0–1 random variables ($e_i = 0$ means message M_i doesn't belong to the emotion class E_{c1}, and $e_i = 1$ means message M_i belongs to the emotion class E_{c1}). Emotion aggregator uses Eq. 5 to measure public emotion over a period of time. Based on Eq. 5, public emotion within the temporal window W is defined as below:

$$E_{\text{public}}(T_c - W, T_c, L, E_{c1}) = \sum_{i=1\ldots n} F(M_i, E_{c1}) \tag{9}$$

where $F(M_i, E_{c1})$ is an indicator function of E_{c1} and n is the number of tweets posted within W. As we know Hoeffding's inequality provides an upper bound on the probability that the sum of random variables deviates $\lambda > 0$ from its expected value as shown by Eq. 10:

$$Pr[|X - \mu| >= \lambda] <= 2e^{-2\lambda^2/n} \tag{10}$$

where X is the sum of independent random variables $X_1, X_2, \ldots, X_n$, with $E[X_i] = p_i$, and the expected value $E[X] = \sum_{i=1\ldots n} p_i = \mu$.

According to the Central Limit Theorem, if n is large then X approaches a normal distribution. We can use Hoeffding's inequality to define an upper bound on the probability that the public emotion E_{c1} deviates from its expected value. Using the Hoeffding bound, for any $\lambda > 0$ we have:

$$Pr[|E_{\text{public}}(T_c - W, T_c, L, E_{c1}) - \mu_e| >= \lambda]$$
$$<= 2e^{-2\lambda^2/n} \qquad (11)$$

where μ_e is the expected number of tweets belong to the emotion class E_{c1} in window W and n is the number of tweets posted within W. Given that n is large in the temporal window W, emotion class E_{c1} can be approximated using a normal distribution.

$$\mu_e = n \times P_e$$

where P_e is the expected rate of the emotion class E_{c1}.

We use the historical average rate of each emotion class as expected rate P_e for that emotion class. For example, a weekly window can be used to average the rate of each emotion class based on all tweets in general. Therefore, other than a sliding detection window over the recent tweets posted about a topic, we also utilize a larger reference window to summarize the information about the tweets posted in general. In fact, our emotion burst detection methodology utilizes two sliding windows. One small window that keeps the rate of each emotion class based on the most recent tweets posted about a topic. Another large reference window that keeps the average rate of each emotion class based on all the past tweets posted in general.

Now we describe our methodology to automatically discover emotion bursts during a real-life event. First, we create an emotion stream by applying EmotexStream system to classify tweets arriving in a stream based on a predefined set of emotion classes. As a second step, our emotion burst detection algorithm then aggregates the tweets of each emotion class into a time-based histogram, using the function in Eq. 5. This aggregation allows us to count the rate of each emotion class in each time period. We then define a sliding window W_{topic} (e.g., daily) over the stream of tweets about a topic aggregated in temporal bins. We also define a large (e.g., weekly) window W_{general} over the general stream of tweets to keep track of the average rate of each emotion class. In order to perform the burst detection, we continuously monitor the rate of public emotion for each emotion class within each temporal window W_{topic}. Whenever the rate of an emotion class exceeds the upper threshold β or falls beneath the lower limit α, an emotion burst is marked as an important moment by keeping its time of occurrence and if it is an up or down case. Then the system signals the occurrence of the detected moments.

4 Experimental results on classifying streams of Twitter messages

We run three separate experiments including, the offline model training, the online classification and the emotion burst detection. In the offline experiment, we collect enough labeled data to build emotion classifiers as described in Sect. 3.1. During the online experiment, we apply our emotion classifiers to classify the live streams of tweets using EmotexStream system (see Sect. 3.2). In the last experiment we select a real-life event and detect the emotion-intensive moments using our method described in Sect. 3.3.

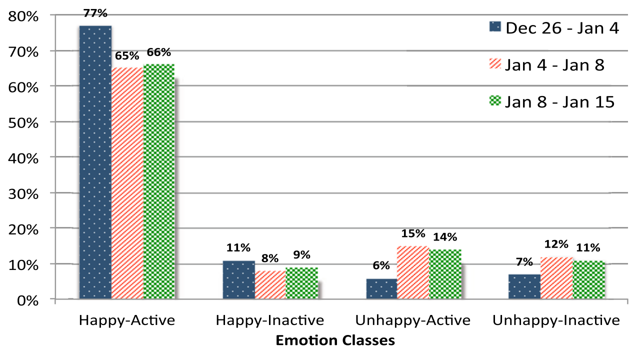

Fig. 7 Distribution of four classes of emotion in collected tweets during and after the new year vacation

4.1 Offline model training: collecting labeled data and building the emotex classifier

To collect emotion-labeled data, we first identify a list of emotion hashtags as explained in Sect. 3.1.1. Using the list of keywords from the Circumplex model (see Fig. 1), a set of emotion hashtags for each class was obtained. Then, we searched for the tweets containing these emotion hashtags and found more emotion hashtags from these tweets, such as the tag "#ifeelsad". At the end, a set of 20 unique emotion hashtags was collected for each emotion class. The objective was to assure that the tags of each class constitute emotions which are different compared with the emotions of the other classes. Using the identified hashtags, labeled data was collected for three weeks between December 26 and January 15. We used Twitter Stream API to collect data from online stream of tweets, which contains a 1% random sample of all tweets. Figure 7 presents the distribution of four classes of tweets that we labeled using hashtags during and after the new year vacation. It shows that the number of happy tweets after vacation is less than the number of happy tweets during vacation by about 13%. More interestingly, the number of unhappy tweets after vacation is more than twice the number of unhappy tweets during vacation. It also shows that the number of active tweets during the vacation are higher than the number of active tweets after vacation by about 4%.

To train our emotion classifiers we select equal size random samples for each emotion class from our collected labeled tweets. In fact, we do random under-sampling to create a balanced training dataset with equal number of samples

Table 1 Number of tweets collected as labeled data

Class	Happy-Active	Happy-Inactive	Unhappy-Active	Unhappy-Inactive	Total
#Tweets before preprocessing	40,000	41,000	44,000	41,000	166,000
#Tweets after preprocessing	34,000	30,000	37,000	34,000	13,5000

Table 2 Distribution of features in the collected data

Features	Happy icon	Sad icon	Angry icon	Sleepy icon	Negation	Punctuation
#Tweets with a feature	5800	1320	1020	270	9050	19450
%Tweets with a feature (%)	4.3	1	0.7	0.2	6.7	14.5

Table 3 Precision, recall and F-Measure ($\beta = 1$) of SVM, Naive Bayes and decision tree using different features

Features	Naive Bayes			SVM			Decision tree		
	Prec.	Rec.	FM	Prec.	Rec.	FM	Prec.	Rec.	FM
Unigram	87.7	86.3	86.3	90.3	89.7	**90**	89.6	89.5	89.5
Unigram emoticon	87.6	86.4	86.4	89.3	88.8	89	89.7	89.6	**89.6**
Unigram punctuation	87.1	86.6	86.6	90.4	89.3	**89.9**	89.8	89.7	89.7
Unigram negation	87.9	86.9	86.9	89.5	88.8	89.1	89.9	89.6	**89.7**
All-features	87.3	87	86.9	90.2	89.5	89.9	90.1	89.9	**90**

in each class [33]. The number of samples in each emotion class is large enough to train classifiers. Table 1 represents the number of labeled tweets selected for each class before and after preprocessing. The removal of noisy tweets during preprocessing decreased the number of tweets by 19%. We explore the usage of different features (see Sect. 3.1.2). Table 2 lists the distribution of features in the collected data after preprocessing.

As described in Sect. 3.1.3 we utilize two types of classification including soft and hard classification. The emotion classification results using soft and hard classification are elaborated below.

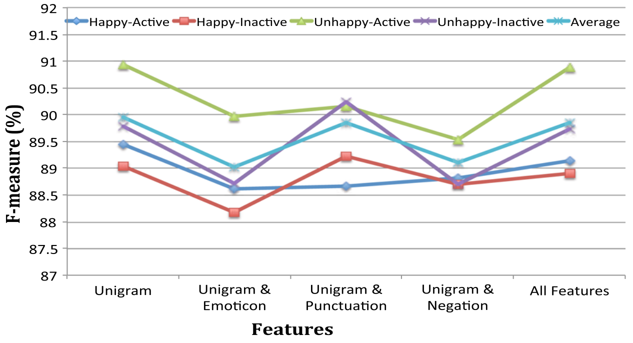

Fig. 8 The F-measure ($\beta = 1$) of SVM model to classify four emotion classes using different features

4.1.1 Emotex: hard classification results

We used two folds of our labeled data to train classifiers and one fold for testing. We used WEKA to train Naive Bayes, and decision tree models and we used SVM-light [34] with a linear kernel to train the SVM classifier.

The classification results are evaluated in terms of F-measure ($\beta = 1$), defined as:

$$2 \times \text{precision} \times \text{recall} \,/\, (\text{precision} + \text{recall})$$

Table 3 presents precision, recall and F-measure of Naive Bayes, decision tree, and SVM using different features based on 3-fold cross-validation. As it shows, decision tree achieved the highest accuracy using all the features. SVM achieved the highest accuracy using unigrams only, while Naive Bayes achieved the highest accuracy using unigrams and negations. Although a decision tree classifier provides high accuracy, it is slow. Therefore, it is not practical for big datasets. SVM-light [34] runs fast and provides the highest accuracy.

The F-measure ($\beta = 1$) values of the SVM model in classifying four emotion classes using different features are presented in Fig. 8. Class unhappy-active got the highest F-score. The active classes (i.e., happy-active and unhappy-active) achieved the highest F-score using unigrams only. However, for the other classes, the highest F-score is achieved using unigrams and punctuations. Across all emotion classes, the unigram-trained model gave the highest overall F-score, and among other features punctuations performed second best.

4.1.2 Emotex: soft classification results

We utilize a probabilistic classifier to measure the soft label based on the probability of assigning a tweet to each emotion class. In this experiment, we run Naive Bayes classifier on our training dataset and produce the class membership probabilities for each tweet. Then the tweets whose maximum probability are higher than a predefined threshold are classified to the class with the maximum probability. The probability threshold is a tuning parameter of the system.

We use the test set ROC curve to find a good probability threshold by resampling. For instance, the tweet "I can live for months on a good compliment." is 65% likely to belong to the happy-active class, 18% likely to belong to the happy-inactive class, 9% likely to belong to the unhappy-active class, and 8% likely to belong to the unhappy-inactive class. Since the maximum probability of this tweet is 65%, it therefore can be classified as a happy-active tweet. As another example, the tweet “Miss you already!” is 19% likely to belong to the happy-active class, 24% likely to belong to the happy-inactive class, 25% likely to belong to the unhappy-active class, and 32% likely to belong to the unhappy-inactive class. The maximum probability of this tweet is 32%, which is fairly small. Thus the tweet cannot be classified with a high enough certainty to render a hard classification.

Figure 9 shows the results of running Naive Bayes classifier on our labeled data with the probability threshold of 50% (Table 1 provides details about our labeled data). As it shows 81% of tweets are classified with the maximum probability higher than the threshold of 50%, where a hard label will be emerged by our system. Only 4% of these tweets are classified wrongly. However, 52% of the tweets whose maximum probability are lower than the threshold are classified inaccurately. In fact, tweets with low confident classification make an error rate of 52%, thus no hard label will be recommended by the system. The results confirm the fact that if tweets are classified with low certainty (i.e., low maximum probability), the classification results have a high error rate. This justifies our approach of forcing a hard classification only for a certain level of confidence.

Based on our observation shown in Fig. 9 tweets with low maximum probability have a higher error rates, we thus separate them in our analysis. In fact, we only consider tweets that are classified with high maximum probability in our analysis. Table 4 shows the accuracy of classification before and after filtering out the tweets with maximum probability lower than the threshold of 50%. As it shows the accuracy has increased by 9.8%, after filtering out the tweets whose maximum probability scores are lower than the threshold of 50%.

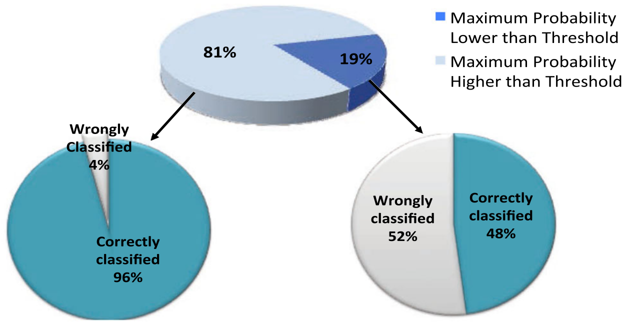

Fig. 9 Distribution of classified tweets based on maximum probability with threshold = 50%

4.1.3 Comparing emotex with the lexical approaches

Existing methods for text classification can be categorized into two main groups: lexical methods and supervised learning methods [23]. To further benchmark the performance of Emotex in classifying emotional messages, we compare it with the lexical approach.

The lexical approach has been previously studied in the context of emotion classification [2,22,35–37]. Lexical methods classify the emotion expressed in texts based on the occurrence of certain words. A lexicon of emotion words is created, in which each word belongs to an emotion class. Text messages are then classified using this emotion lexicon, typically by employing frequency counts of terms. The lexical methods may consider only terms of the lexicon directly or may associate numerical weights with these terms.

Lexical methods are based on shallow word-level analysis, and can recognize only surface features of the text. They usually ignore semantic features (e.g., negation) [23]. Moreover they rely on an emotion lexicon, which is difficult to construct a comprehensive set of emotion keywords. The creation of emotional lexicon is both time-consuming and labor-intensive, and requires expert human annotators.

A variety of Emotion lexicons including ANEW lexicon [26], WordNet Affect [38], and LIWC dictionary [27] have been developed. To compare the results of Emotex with the lexical approach we utilize ANEW lexicon, which contains 2477 affect words that are rated for valence and arousal on a 1–9 scale. To classify messages using ANEW lexicon, the average valence and arousal of each message is estimated using the following formulas:

$$\text{Valence}_{\text{tweet}} = \frac{\sum_{i=1}^{n} v_i f_i}{\sum_{i=1}^{n} f_i} \tag{12}$$

$$\text{Arousal}_{\text{tweet}} = \frac{\sum_{i=1}^{n} a_i f_i}{\sum_{i=1}^{n} f_i} \tag{13}$$

Table 4 Classification results of Naive Bayes after removing tweets with low maximum probability

	#Tweets	Precison (%)	Recall (%)	F-measure (%)
No Filtering	134,100	88	83.7	86
Removing tweets with low max-probability	108,516	96	94.4	95.8

Table 5 Comparing the classification results of Emotex with ANEW lexical approach based on precision, recall and F-measure ($\beta = 1$)

Emotion classes	Emotex			Lexical method		
	Prec.	Rec.	FM	Prec.	Rec.	FM
Happy-Active	84.2	95.4	89.5	54.4	60.6	57.3
Happy-Inactive	94.3	84.4	89.1	64.2	58.5	61.2
Unhappy-Active	91.4	90.5	91	79.4	44	56.6
Unhappy-Inactive	91.2	88.4	89.8	91.5	52.5	66.7

where n is the number of affect words occurring in the tweet, f_i is the frequency of the ith affect word, and v_i and a_i are the valence and arousal of the ith affect word, respectively.

Then using the following heuristic the message can be easily classified: Less than 5 means low arousal/valence, more than 5 means high arousal/valence, and equal to 5 is neutral. For example, the tweet "Family and friends made this Christmas great for me." with the affect words family, friends and christmas, the valence and arousal values are as following:

$$\text{Valence} = (7.74 + 7.65 + 7.8) \; / \; 3 \; = 7.73$$

$$\text{Arousal} = (5.74 + 4.8 + 6.27) \; / \; 3 \; = 5.60$$

Since both valence and arousal are larger than five, the tweet is labeled as happy-active. We compare the performance of Emotex with the lexical approach in classifying our labeled data shown in Table1. Table 5 lists the classification results of Emotex and the lexical approach in classifying different emotion classes. As the table shows, the F-score of Emotex is about 30% higher than the lexical approach utilizing ANEW lexicon.

4.2 Online classification: classifying live streams of tweets

After building the Emotex system as described in Sect. 4.1, we now deploy it to classify emotion in live streams of tweets. For this purpose, we develop EmotexStream framework presented in Sect. 3.2. Based on the EmotexStream system, we first detect emotion-present tweets and separate them from emotion-absent tweets. Therefore, we utilize our binary classifier developed using several emotion lexicons. For the binary classification experiment, we collect a large amount of general tweets from USA without filtering them by any specific hashtag or keyword (see Table 6). After cleaning up the noise, we classify them using our binary classifier. Emotion-present tweets will then go through the feature selection and multi-class emotion classifier generated by our Emotex system to classify them based on our defined classes of emotion. Table 6 shows the results of our binary classification experiment. It is interesting to observe that in a random sample of tweets the majority does in fact contain identifiable emotion.

We also evaluate the accuracy of our binary classifier through a user study. We randomly select a sample set of general tweets including 50 tweets from the dataset described in Table 6. Then we ask 25 graduate students to manually classify them. They classified each tweet into two groups namely, emotion-present tweets versus emotion-free tweets (i.e., tweets with explicit emotion versus tweets without any emotion). Fleiss-Kappa for the labelers is 0.28 which shows a fair agreement. The manual label of each sample tweet is selected based on the majority votes of labelers for that tweet. There were three tweets which didn't receive the absolute majority of the votes. We didn't consider them in our evaluation. After creating manual labels, we classified the selected sample tweets using our binary classifier and compared them with manually classified results. The manual labels served as the ground truth labels. We generated our binary classifier results using two different lexicons LIWC and ANEW.

Table 7 shows the precision, recall and F-measure ($\beta = 1$) of the binary classifier through comparison with the manual classification. As the results show, using a larger lexicon (i.e., LIWC and ANEW) increased recall and F-measure, compared with using only one lexicon. Therefore, for the binary classification task we use a multi-lexicon by combining dif-

Table 6 Results of binary classification in live stream of tweets

	Total tweets	After preprocessing	Emotion tweets	No-emotion tweets
Number	105,134	104,924	56,472	48,452
Percent	100%	99.8%	53.7%	46.1%

Table 7 Evaluating binary classification results by comparing them with manually classification results

Lexicon	Precision (%)	Recall (%)	F-Measure (%)
LIWC	93	77	84.3
ANEW	74	82	77.8
LIWC& ANEW	78	95	85.6

ferent lexicons. A larger lexicon increases recall, but may decrease precision.

4.3 Detecting emotion bursts in live tweet streams

Using EmotexStream, we are able to classify live streams of tweets in real-time. We now use this system to measure and analyze public emotion in a specific location. The objective of this experiment is to observe the temporal distribution of crowd emotion and detect important moments during the real-life events. We select the death of Eric Garner in New York [1] which stirred public protests and rallies with charges of police brutality. Eric Garner died after a police officer put him in a choke-hold, which caused many discussions on social media. On December 3, 2014, a grand jury decided not to indict the police officer. We utilize the Twitter search API to search for tweets containing a specified set of hashtags. We collected 4K tweets containing the hashtag "Garner" from November 24, 2015 until January 5, 2015 posted in New York. After collecting tweets, we classify them using our EmotexStream model (see Sect. 3.2). Then, the emotion-classified tweets are aggregated into a daily-based histogram. Finally, using the methodology described in Sect. 3.3 we measure public emotion and detect emotion-critical moments.

Figure 10 presents the temporal changes of different classes of emotion in New York during the selected event. The important moments are also specified in this figure. The distribution shows a predominance of sad and angry emotions over happy emotion in many days during the event. In order to predict the important moments as emotion bursts, we apply a sliding window W_{event} of length one day over the emotion stream of tweets aggregated in daily bins, as described in Sect. 3.3. Also a reference weekly window W_{general} is applied over the general stream of tweets to calculate the average rate of each emotion class. Then, we continuously monitor the frequency rate $E_{\text{public}}(Tc - W_{\text{event}}, Tc, L, E_{c1})$ over time for each emotion class E_{c1}. Whenever this rate for an emotion class exceeds the upper threshold of β or falls beneath the lower limit α, an emotion burst is reported. Table 8 presents the days of abrupt changes in happiness. The second row shows the frequency rate of emotion bursts which are out of range. The last row shows the low and high boundaries. Comparing the results of this table with the important moments specified in Fig. 10 confirms that our method is able to detect emotion-critical moments.

5 Evaluating the emotex labeling method

In the preceding sections, we have assumed that hashtags are true labels of the emotions expressed in text messages. However, the question still remains whether this assumption is correct. To answer this question, we need to determine whether human annotators would categorize texts into the same emotion classes selected by automatic labeling using hashtags. To evaluate the accuracy of hashtags as emotion labels, we performed two user studies in which two different classes of annotators participated. First, psychology experts (counselors and psychology graduate students) and then psychology novices (the crowd) were asked to classify texts into emotion classes.

5.1 Comparing hashtag labels with crowdsourced labels

This user study compares the accuracy of emotion labels that are created automatically using hashtags with labels made by nonexpert annotators (the crowd). We design the study by randomly selecting 120 tweets (i.e., 30 tweets from each emotion class) from our collected emotion-labeled tweets (see Sect. 4.1). The tweets are shuffled to make their order random. Any embedded hashtags were removed from these 120 tweets as they are to serve as potential labels. Then the participants were asked to indicate the emotion expressed in each message by selecting the pleasure level (high for happy or low for unhappy), and the arousal level (high for active or low for inactive). We recruited labelers from the students in an introductory psychology class at Worcester Polytechnic Institute. Our user study was run online using the Qualtrics [2] survey system for three months. Sixty students participated and 49 students completed the survey.

The perception of emotions expressed in texts tends to be subjective and diverse. As expected, inconsistencies occurred in the answers, such that in some cases different participants categorized the same text into different classes. Thus, we measure to what degree the participants agreed on the level of pleasure or activation of each tweet. We utilized Fleiss-Kappa to measure the level of agreement between a fixed number of labelers in classifying subjects. The Fleiss-Kappa value for inter-labeler agreement of the pleasure level of tweets was 0.67, which corresponds to a substantial agreement. This value for the activation level was 0.25 which

[1] https://en.wikipedia.org/wiki/Death_of_Eric_Garner.

[2] http://www.qualtrics.com.

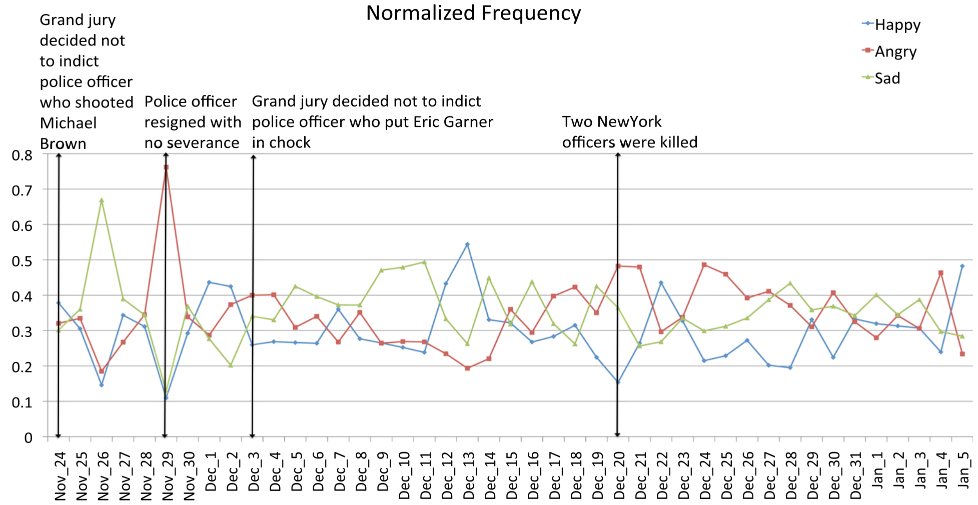

Fig. 10 Changes of emotion about selected sad events in New York

Table 8 Detected burst changes in happiness

Date	Nov 26	Nov 29	Dec 19	Dec 20	Dec 27	Dec 28	Dec 30
Happy rate	210	175	576	462	463	360	503
Boundary(α, β)	(360, 936)	(400, 1040)	(641, 1668)	(753, 1957)	(573, 1491)	(461, 1199)	(561, 1459)

shows a low level of agreement. In summary, although the annotators substantially agreed on the level of pleasure, there was a relatively low agreement among them for the level of activation. This conclusion can be explained by the fact that authors of text messages tend to express pleasure in explicit and unambiguous terms. For example, the tweet "Final weeks is going to be a death of me!" shows sadness. However, it does't clearly indicate the level of arousal (i.e., activation).

The result of this study indicates that the labels created by nonexperts to classify emotion in texts are not sufficiently reliable. This casts doubt on the use of the crowd (i.e., via Amazon Mechanical Turk) for this particular task of emotion classification. Note that participants in our study are a relatively notable crowd, as they are students in psychology that are trained to do user studies and have a general interest.

5.2 Comparing hashtag labels with expert labels

As the results of previous study indicates the level of agreement among crowd labelers is not sufficient to be able to consider them as ground truth especially for evaluating hashtag labels. Instead we sought the help of domain experts for labeling. We asked three psychology experts to manually label 120 tweets (the same set of tweets that had been utilized in Sect. 5.1). One of the experts is the director of counseling at WPI Student Development and Counseling Centre. The other two experts are graduate students in psychology who have been trained to classify emotions.

Table 9 Comparing Fleiss-Kappa values of crowd and expert labelers

Labeler	Pleasure level	Activation level
Crowd Labeler	0.67	0.25
Expert Labeler	0.84	0.64

The Fleiss-Kappa measure of agreement between experts for the pleasure level of tweets is 0.84 which constitutes a high level of agreement. This value for the activation level is 0.64 which shows a substantial agreement. Table 9 lists the Fleiss-Kappa values of crowd labelers versus expert labelers. The agreement between experts is much higher than the agreement between crowd labelers. These results indicate that emotion labeling by trained experts is more reliable. It thus is more appropriate to be utilized as the ground truth. However, we note that if experts are used to label messages, crowdsourcing will be prohibitively expensive.

We now utilize the expert labels to evaluate the accuracy of hashtags. Table 10 lists the accuracy of hashtags based on expert labels. Hashtag labels are same as expert labels in 102 tweets. There are 14 tweets for which their hashtag labels are different from the expert labels. Also there is no consensus among experts about 4 tweets. Therefore, in about 88% of the cases, emotions indicated by hashtags embedded in tweets accurately captured the author's emotion indicated by the ground truth (i.e., expert labels). Most of the mismatches between hashtags and expert labels belong to the arousal level

Table 10 Accuracy of hashtag labels based on expert labels

Expert	Counseling director	Trained expert1	Trained expert2	Experts consensus
Accuracy	81%	81%	84%	**88**%

of tweets (i.e., active or inactive), which is not an intuitive concept to understand by nonpsychologists.

6 Related work on emotion detection in text

This section briefly surveys prior work on classifying emotion in texts. Emotion detection methods can be divided into lexicon-based methods and machine learning methods.

6.1 Lexicon-based methods

Most research on textual emotion recognition is based on building and employing emotion lexicons [22,35,36]. Lexicon-based methods rely on lexical resources such as lexicons, set of words or ontologies. They usually start with a small set of seed words. Then they bootstrap this set through synonym detection or online resources to collect a larger lexicon. Ma et al. [35] searched WordNet for emotional words for all 6 emotional types defined by Ekman [19]. They then assigned weights to those words according to the proportion of Synsets with emotional association that the words belong to. Strapparava and Mihalcea [22] constructed a large lexicon annotated for six basic emotions: anger, disgust, fear, joy, sadness and surprise. They used linguistic information from WordNet Affect [38].

In another work, Choudhury et al. [2] identified a lexicon of more than 200 moods frequently observed on Twitter. Inspired by the Circumplex model, they measured the valence and arousal of each mood using mechanical turk and psychology literature sources. Then, they collected posts which had at least one of the moods in their mood lexicon as indicated by a hashtag at the end of a post.

Mohammed et al. [39] and Wang et al. [1] collected emotion-labeled tweets using hashtags for several basic emotions including joy, sadness, anger, fear, and surprise. They showed through experiments that emotion hashtags are relevant and match with the annotations of trained judges. Canales et al. also collected emotion-labeled corpora using a bootstrapping process [40]. They annotated sentences from blogs posts based on the Ekman's six basic emotions [19].

Recently, researchers have explored social media such as Twitter to investigate its potential to detect depressive disorders. Park *et al.* [5] ran studies to capture the depressive mood of users in Twitter. They studied 69 individuals to understand how their depressive states are reflected in their tweets. They found that people post about their depression and even their treatments on social media. Their results showed that participants with depression exhibited an increased usage of words related to negative emotions and anger in their tweets. Another effort for emotion analysis on Twitter data was undertaken by Bollen et al. [20]. They extracted 6 basic emotions (tension, depression, anger, vigor, fatigue, confusion) using an extended version of POMS (Profile of Mood States). They found that social, political, cultural and economic events have a significant and immediate effect on the public mood.

6.2 Machine learning-based methods

Machine Learning methods apply statistical algorithms on linguistic features, which can be supervised or unsupervised. A few researchers applied supervised learning methods to identify emotions in texts. Choudhury et al. [4] detected depressive disorders by measuring behavioral attributes including social engagement, emotion, language and linguistic styles, ego network, and mentions of antidepressant medication. Then they leveraged these behavioral features to build a statistical classifier that estimates the risk of depression. They crowdsourced data from Twitter users who have been diagnosed with mental disorders. Their models showed an accuracy of 70% in predicting depression.

Another work accomplished by Qadir et al. [41] to learn lists of emotion hashtags using a bootstrapping framework. Starting with a small number of seed hashtags, they trained emotion classifiers to identify and score candidate emotion hashtags. They collected hashtags for five emotion classes including affection, anger, anxiety, joy and sadness.

Purver et al. [21] tried to train supervised classifiers for emotion detection in Twitter messages using automatically labeled data. They used the 6 basic emotions identified by Ekman [19] including happiness, sadness, anger, fear, surprise and disgust. They used a collection of Twitter messages, all marked with emoticons or hashtags corresponding to one of six emotion classes, as their labeled data. Their method did better for some emotions (happiness, sadness and anger) than others (fear, surprise and disgust). Their work is similar to ours, however they used categorical emotion models and their overall accuracies (60%) were much lower than the accuracy achieved by our approach.

Another supervised learning work with categorical emotion models is done by Suttles and Ide [42]. They classify emotions according to a set of eight basic bipolar emotions defined by Plutchick including anger, disgust, fear, happiness, sadness, surprise, trust and anticipation. This allows them to treat the multi-class problem of emotion classification as a binary problem for opposing emotion pairs.

An unsupervised method was proposed by Agrawal and An [43]. They presented an unsupervised context-based approach based on a methodology that does not depend on any existing affect lexicon; therefore, their model is flexible to classify texts beyond Ekman's model of six basic emotions. Another unsupervised approach was developed by Calvo et al. [24]. They proposed an unsupervised method using dimensional emotion model. They used a normative database ANEW [26] to produce tree-dimensional vectors (valence, arousal, dominance) for each document. They also compared this method with different categorical approaches. For the categorical approaches three dimensionality reduction techniques: Latent Semantic Analysis (LSA), Probabilistic Latent Semantic Analysis (PLSA) and Nonnegative Matrix Factorization (NMF) were evaluated. Their experiments showed that the categorical model using NMF and the dimensional model tend to perform best.

7 Conclusion

In this paper, we study the problem of automatic emotion detection in text stream messages. We develop and evaluate a supervised machine learning system to automatically classify emotion in text streams. Our approach includes two main tasks: an offline training task and an online classification task. We develop a system called Emotex to create models for classifying emotion during the first task. Our experiments show that created models correctly classify emotion in 90% of text messages. For the second task, we develop a two-stage framework called EmotexStream to classify live streams of tweets for the real-time emotion tracking. First it creates a binary classifier to separate tweets with explicit emotion from tweets without emotion. Then it conducts a fine-grained emotion classification on tweets with explicit emotion using Emotex. Moreover, we propose an online method to measure public emotion and detect emotion-intensive moments in live streams of text messages.

To address the problem of fuzzy boundary and variations in expression and perception of emotions, a dimensional emotion model is utilized to define emotion classes. Furthermore, a soft (fuzzy) classification approach is proposed to measure the probability of assigning a message into each emotion class.

References

1. Wang, W., Chen, L., Thirunarayan, K., Sheth, AP.: Harnessing twitter big data for automatic emotion identification. In: 2012 International Conference on Social Computing (SocialCom), pp 587–592. IEEE (2012)
2. De Choudhury, M., Counts, S., Gamon, M.: Not all moods are created equal! exploring human emotional states in social media. In: ICWSM'12 (2012)
3. Wakamiya, S., Belouaer, L., Brosset, D., Lee, R., Kawai, Y., Sumiya, K., Claramunt, C.: Measuring crowd mood in city space through twitter. In: International Symposium on Web and Wireless Geographical Information Systems, pp 37–49. Springer (2015)
4. Choudhury, MD., Gamon, M., Counts,S., Horvitz, E.: Predicting depression via social media. In: ICWSM'13, The AAAI Press (2013)
5. Park, M., Cha, C., Cha, M .: (2012) Depressive moods of users portrayed in twitter. In: Proceedings of the ACM SIGKDD Workshop on Healthcare Informatics, HI-KDD
6. Guthier, B., Alharthi, R., Abaalkhail, R., El Saddik A.: Detection and visualization of emotions in an affect-aware city. In: Proceedings of the 1st International Workshop on Emerging Multimedia Applications and Services for Smart Cities, pp 23–28. ACM (2014)
7. Resch, B., Summa, A., Zeile, P., Strube, M.: Citizen-centric urban planning through extracting emotion information from twitter in an interdisciplinary space-time-linguistics algorithm. Urban Plann. **1**(2), 114–127 (2016)
8. Kanhabua, N., Nejdl, W.: (2013) Understanding the diversity of tweets in the time of outbreaks. In: Proceedings of the 22nd international conference on World Wide Web companion, International World Wide Web Conferences Steering Committee, pp. 1335–1342
9. Hasan, M., Agu, E., Rundensteiner, E.: (2014) Using hashtags as labels for supervised learning of emotions in twitter messages. In: Proceedings of the ACM SIGKDD Workshop on Healthcare Informatics, HI-KDD
10. Go, A., Bhayani, R., Huang, L.: Twitter sentiment classification using distant supervision. CS224N Project Report, Stanford, pp 1–12 (2009)
11. Pak, A., Paroubek, P.: Twitter as a corpus for sentiment analysis and opinion mining. In: Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10), ELRA, Valletta, Malta (2010)
12. Barbosa, L., Feng, J.: Robust sentiment detection on twitter from biased and noisy data. In: Proceedings of the 23rd ACL: Posters, Association for Computational Linguistics, pp 36–44 (2010)
13. Kouloumpis, E., Wilson, T., Moore, J.: Twitter sentiment analysis: The good the bad and the omg! In: ICWSM'11, The AAAI Press (2011)
14. Gunes, H., Schuller, B., Pantic, M., Cowie, R.: Emotion representation, analysis and synthesis in continuous space: A survey. In: 2011 IEEE International Conference on Automatic Face & Gesture Recognition and Workshops (FG 2011), pp. 827–834. IEEE (2011)
15. Wang, X., Wei, F., Liu, X., Zhou, M., Zhang, M.: Topic sentiment analysis in twitter: a graph-based hashtag sentiment classification approach. In: Proceedings of the 20th ACM international conference on Information and knowledge management, pp 1031–1040. ACM (2011)
16. Russell, J.A.: A circumplex model of affect. J. Personal. Soc. Psychol. **39**, 1161–1178 (1980)
17. Hasan, M., Rundensteiner, E., Agu, E.: Emotex: Detecting emotions in twitter messages. In: Proceedings of the Sixth ASE International Conference on Social Computing (SocialCom 2014), Academy of Science and Engineering (ASE), USA (2014)
18. Russell, J.A., Barrett, L.F.: Core affect, prototypical emotional episodes, and other things called emotion: dissecting the elephant. J. Personal. Soc. Psychol. **76**(5), 805 (1999)
19. Ekman, P.: Basic emotions. Handb. Cognit. Emot. **98**, 45–60 (1999)

20. Bollen, J., Mao, H., Pepe, A.: Modeling public mood and emotion: Twitter sentiment and socio-economic phenomena. In: ICWSM'11 (2011)
21. Purver, M., Battersby, S.: Experimenting with distant supervision for emotion classification. In: Proceedings of the 13th EACL, Association for Computational Linguistics, pp. 482–491 (2012)
22. Strapparava, C., Mihalcea, R.: Learning to identify emotions in text. In: Proceedings of the 2008 ACM symposium on Applied computing, pp. 1556–1560. ACM (2008)
23. Liu, H., Lieberman, H., Selker, T.: A model of textual affect sensing using real-world knowledge. In: Proceedings of the 8th international conference on Intelligent user interfaces, pp. 125–132. ACM (2003)
24. Calvo, R.A., Mac Kim, S.: Emotions in text: dimensional and categorical models. Computat. Intell. **29**(3), 527–543 (2013)
25. Princeton, U.: (2010) Wordnet. http://wordnet.princeton.edu
26. Bradley, M.M., Lang, P.J.: Affective norms for english words (anew): Instruction manual and affective ratings. In: Technical Report Citeseer (1999)
27. Pennebaker, JW., Francis, ME., Booth, RJ.: Linguistic inquiry and word count: Liwc 2001. Mahway: Lawrence Erlbaum Associates p. 71 (2001)
28. rup Nielsen, F.: A new anew: evaluation of a word list for sentiment analysis in microblogs. In: Proceedings of the ESWC2011 Workshop on 'Making Sense of Microposts': Big things come in small packages, vol. 718, pp. 93–98 (2011)
29. Liu, Y., Zhang, H.H., Wu, Y.: Hard or soft classification? Large-margin unified machines. J. Am. Stat. Assoc. **106**(493), 166–177 (2011)
30. Zadrozny, B., Elkan, C.: Transforming classifier scores into accurate multiclass probability estimates. In: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 694–699. ACM (2002)
31. Platt, J., et al.: Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. Adv. Large Margin Classif. **10**(3), 61–74 (1999)
32. Hasan, M., Rundensteiner, E., Kong, X., Agu, E.: Using social sensing to discover trends in public emotion. In: 2017 IEEE 11th International Conference on Semantic Computing (ICSC), pp. 172–179. IEEE (2017)
33. Branco, P., Torgo, L., Ribeiro, R.P.: A survey of predictive modeling on imbalanced domains. ACM Comput. Surv. (CSUR) **49**(2), 31 (2016)
34. Joachims, T.: Making large-scale SVM learning practical. In: Schölkopf, B., Burges, C.J., Smola, A. (eds.) Advances in Kernel Methods-Support Vector Learning. MIT Press, Cambridge (1999)
35. Ma, C., Prendinger, H., Ishizuka, M.: Emotion estimation and reasoning based on affective textual interaction. In: Affective Computing and Intelligent Interaction, pp. 622–628. Springer (2005)
36. Neviarouskaya, A., Prendinger, H., Ishizuka, M.: Textual affect sensing for sociable and expressive online communication. In: Affective Computing and Intelligent Interaction, pp. 218–229. Springer (2007)
37. Dodds, P.S., Danforth, C.M.: Measuring the happiness of large-scale written expression: songs, blogs, and presidents. J. Happiness Stud. **11**(4), 441–456 (2010)
38. Strapparava, C., Valitutti, A.: Wordnet affect: an affective extension of wordnet. In: Proceedings of 4th International Conference on Language Resources and Evaluation, LREC, vol 4, pp. 1083–1086 (2004)
39. Mohammad, SM.: # emotional tweets. In: Proceedings of the First Joint Conference on Lexical and Computational Semantics, Association for Computational Linguistics, pp. 246–255 (2012)
40. Canales, L., Strapparava, C., Boldrini, E., Martnez-Barco, P.: Exploiting a bootstrapping approach for automatic annotation of emotions in texts. In: 2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA), pp. 726–734. IEEE (2016)
41. Qadir, A., Riloff, E.: Bootstrapped learning of emotion hashtags# hashtags4you. WASSA **2013**, 2 (2013)
42. Suttles, J., Ide, N.: Distant supervision for emotion classification with discrete binary values. In: International Conference on Intelligent Text Processing and Computational Linguistics, pp. 121–136. Springer (2013)
43. Agrawal, A., An, A .: Unsupervised emotion detection from text using semantic and syntactic relations. In: Proceedings of the The 2012 IEEE/WIC/ACM International Joint Conferences on Web Intelligence and Intelligent Agent Technology-Volume 01, pp. 346–353. IEEE Computer Society (2012)