Emotion Recognition Using Mobile Phones

Shams Shapsough, Ahmed Hesham, Youssef Elkhorazaty, Imran A. Zualkernan, and Fadi Aloul

Department of Computer Science & Engineering
American University of Sharjah, UAE

***Abstract*—The availability of built-in sensors in mobile phones has enabled a host of innovative applications. One class of applications deals with detecting a user's emotions. Previous applications have primarily relied on recording and displaying self-reported emotions. This paper presents an intelligent emotion detection system for mobile phones implemented as a smart keyboard. The smart keyboard independently infers a user's emotional state using machine learning techniques. The system uses accelerometer readings and various aspect of typing behavior like speed, number of backspaces, and time delay between letters to train a classifier to predict emotions. Naïve Bayes, J48, IBK, Multi-response linear regression and SVM were evaluated and J48 was found to be the best classifier with over 90% accuracy and precision. In addition to providing emotive feedback to individual users, the system also uses geo-tagged data to collect and display emotional states of regions or countries through a website.**

Keywords—Emotion recognition, Machine intelligence, Mobile phones, Sensors

I. INTRODUCTION

With the advent of computing came a growing dependency on mobile phones that went beyond the communication purpose they were originally intended for. People today use mobile phones to carry out a range of daily tasks like shopping, banking, socializing, ordering food, etc. In addition, mobile phones are also being used as safety [1] and medical devices [2, 3]. Over time, mobile phones have increasingly become more complex to meet consumer's demands and to satisfy an ever growing need for more computational power. An average mobile phone now comes equipped with communication modules (Bluetooth, Wi-Fi etc.), an arrays of sensors (accelerometers, gyroscopes, temperature sensors etc.) and significant computational power. These built-in sensors can be used to deploy unique applications that were not possible in the past.

A mobile phone user's emotional state can be inferred using on-board sensors [4]. Such technology has applications in a variety of domains. Healthcare is one key area of application where users can keep track of their own emotional health. Such an application enables users to determine, for instance, sudden shifts in mood, or changes in mental health allowing a person to seek help if needed [5]. Finally, through a web service, public users could also collect demographics about the emotional state of a populace. Not only that, but medical organizations can also infer correlations between geographical conditions, context, and psychological wellbeing of individuals in that region.

Emotion recognition on mobile devices has typically relied on two broad approaches. First, users are asked to fill in surveys and/or questionnaires about their emotional state and the mobile phone simply tracks these emotions [6]. A second approach uses language processing to determine user's mood [7]. Filling out forms is cumbersome and, for example, not likely to happen when someone is angry. Similarly, using natural language processing for emotion detection, especially on a mobile phone, is not trivial. For example, if someone were to type "lol" or "rofl" etc., the natural language processor, unless configured to recognize these terms, would infer, falsely, that the user made a spelling mistake. Moreover, if the nature of language is taken into account and the way in which people develop words and short hand notations as new technology comes around (Google is not an English word, but is now used as a verb "let me Google that"), it becomes very difficult to design a system that can consistently detect a user's emotional state based on language alone [7].

This paper proposes to recognize the emotional state of a user by exploiting the various built-in sensors in a mobile phone. This is achieved by creating a soft-keyboard that uses sensor data to eventually determine a user's current emotion. This soft-keyboard replaces the default mobile phone keyboard and can be used with any application running on the mobile phone. The soft-keyboard connects securely to web-service that provides personalized statistics reflecting the emotional state of a user through time [8]. Others can access the web service to view the average emotional profiles of populations across geographical locations.

Rest of the paper is organized as follows. The next section describes previous work in detecting emotions using mobile phones. Section III describes the design of the system including an evaluation of the machine learning algorithms used. Section IV shows the system architecture and implementation. Section V presents the conclusion and future work.

II. BACKGROUND

This section provides a brief overview of previous mobile phone applications that recognize user emotions. The use of machine learning for emotion recognition is also discussed.

A. Detecting Emotions Using Mobile Phones

Shivhari and Saritha [9] proposed key spotting method to classify the user's emotional state based on keywords found in the user's text input. The algorithm uses a six step process that consist of 1) Capturing User's text input, 2) Tokenizing text, 3) Identifying keywords, 4) Analyzing keywords and weighing them on a pre-set scale quantifying the emotion, 5) Adding the weights to create a final classification. There are two primary limitations with this method of emotion classification. First, this method does not account for the context in which the words occur, but merely checks for the occurrence of specific keywords. Secondly, the algorithm does not consider user's word choice patterns as part of the classification process; not considering word choice pattern leads to the output being inaccurate for a wide range of users [10].

EmotionSense [11] is a stand-alone application that works by first asking users to sign up to their web service through an email account. This is done to allow data gathering for later access by the user. After sign-up, the users are taken through a brief survey that asks them questions about their emotions followed by a question that asks users to select the intensity of their current emotions on a graph. For example, the user enters the intensity of moods like "calm" or "anxious." Based on manual input, the application plots the user's mood (positive vs. negative and sleepy vs. alert, for example) on a grid. In addition, the application uses the built-in sensors like the accelerometer and the GPS to determine if the user is active or not. Level of social interaction is measured by the amount of social media used. Every week, the app unlocks a new method of detecting the user's emotion. For example, in the second week, it unlocks detection using location, then SMS patterns, and so on. Every day, the app asks the user how he feels and adds their emotion to the output grid. This is done to allow the application to develop a baseline against which it can determine the user's emotional state based on phone usage information. The user is able to check his/her statistics at any time. It should be mentioned that moods are self-reported.

T2 Mood Tracker [12] is a stand-alone application that acts like a mood diary by frequently asking the user to rate how he/she feels. This is done through the use of sliders; one for each emotion. The app then plots the emotional data over time. The application allows the user to generate reports on dimensions like anxiety, depression etc. Unlike Emotion Sense, this application does not perform a computational analysis of the user's device usage parameters. The app only determines the user's state from the data he or she provided manually.

B. Using Machine Learning for Emotion Recognition

Many machine learning algorithms attempt to automate text categorization [13]. For emotion recognition, the algorithm needs to classify a user's emotional state (e.g., angry) based on the provided user input (e.g., text being typed, sensor data etc.). The primary advantage of using the machine learning approach is its ability to tailor the classification based on an individual user's behavior. Supervised machine learning algorithms are used to solve this class of problems. Algorithms in this category initially require data input to be labelled with the desired output. After the initial training period the algorithm can begin to classify new input based on the pre-classified data that was originally provided. This paper considers multiple learning algorithms; Naïve Bayes. Support Vector Machines, J48 and Regression.

III. EMOTION DETECTION

A. Approach

The primary idea behind the approach presented in this paper is to collect sensor data while a user is typing on the keyboard. As a user types, he or she is prompted to indicate their current emotional state. In doing so, sensor data from the phone is tagged with the current emotional state of a particular user. Once enough data is collected, machine learning techniques are used to build classifiers that can predict the user's current emotional states based on their current typing behavior. The approach proposed in this paper differs from previous work. First, the proposed system uses a soft-keyboard, and hence can be used with any application as opposed to being tied to a specific application. This has the advantage of applying emotion recognition within the context of any application that uses the keyboard. Secondly, the proposed system is data-centric and automatically collects users' data as they type on the keyboard in any application. Rather than always relying on the user to self-report their emotion, once enough data is collected, the application predicts emotions intelligently based solely on the sensor data from the mobile phone.

The approach works as follows: A user first downloads and installs the soft keyboard application on their mobile phone. After this step, a user can use this keyboard for any application on their mobile device. Every time this keyboard is used, sensor and typing data is collected. A machine learning algorithm subsequently used to construct an emotion classifier based on captured data. In this training phase, the user is asked to indicate their emotion while they type. This recorded emotion is used as a tag, and the tagged data is used as an input train the classifier. After the training stage is over, the classifier takes the current typing behavior of the user and predicts their emotional state. Optionally, the emotional state data can be uploaded to a webserver for public usage.

B. Feature Detection

The primary input to the machine learning algorithm is a set of feature vectors derived from typing behavior of the user. Each feature vector contains features representing a uniform segment of typing behavior. A feature vector consists of average acceleration, average time delay between typed letters, number of backspaces, and the associated user emotion. The first two components are calculated from mobile phone's sensor data, while the number of backspaces is recorded from the keyboard.

C. Classification

Using feature vectors as input, the following machine learning algorithms were evaluated to find the best classification method.

- Naïve Bayes – Estimates the probability of an entry being of a certain class based on previous entries [14, 15].
- J48 Decision Tree – Creates a C4.8 decision tree that splits the data into different subsets [16, 17].
- Lazy IBK – A nearest neighbour approach, where the distance between two feature vectors is calculated and a class is assigned based on the nearest neighbour [18].
- Multi-response linear regression – Classification possibilities are converted into binary and a regression model is created for each possible class [19].
- SVM – Creates a hyper-plane separating the various emotions [20].

D. Implementation

Implementation of each aforementioned classification algorithm was used to evaluate their relative effectiveness. The Weka machine learning toolkit [21] was used for this purpose. The test set used contained 307 feature vectors, where 109 vectors were tagged as 'Neutral', 66 as 'Angry', 84 as 'Happy' and 48 as 'Sad'. The test set was collected over a period of one month from three volunteer users. Ten folds cross-validation was used. Cross validation splits the set into 10 parts. Every iteration uses 9 slices for training and the last slice as a test set. This is repeated until every slice is used a test set. Finally each algorithm iterates the set for the eleventh time using the full set for testing.

E. Results

Receiver Operating Characteristic (ROC) for the various emotional states are shown next. An ROC curve compares the true positive rate or correctly classified instances against the false positive rate or incorrectly classified instances. A perfect classifier has an upside down 'L' shape while the worst classifiers have a diagonal ROC curve. Figure 1 shows ROC curve for 'Neutral' state as opposed to all the other states. As the Figure shows, SVM is the worst performing classifier because its ROC curve is diagonal. The two best performing classifiers for the Neutral state are Regression and J48. The fact that SVM performed the worst and J48 performed the best suggests that it is not possible to construct linear hyper-planes through the data space. Rather, specific hyper cubes within the space represent the various emotional states.

Figure 2 shows the ROC curve for 'Happy' emotion. As the Figure shows, the results are similar to those of 'Neutral' emotion. J48 performs the best while SVM is the worst classifier. Figure 3 shows the decisions tree generated by J48 that was the best classified for the test data set. As can be seen the 'Angry' emotion can detected easily by using only the Accelerometer data. However, a complex set of decisions is required to differentiate "Neutral" from the "Happy" state as shown by the left sub-tree.

Based on ROC curves, it was concluded that, overall J48 performed the best. The summarized results for J48 are shown in Table 1. As the Table shows, both precision and recall are above 90% and the overall F-Measure is also 90% which means that J48 is a good classifier. Table 2 shows the results for SVM clearly indicating that the approach did not work well as Precision, Recall and F-Measure are all below 60%.

In summary, the emotion detection approach uses mobile phone's sensor data and user's typing behavior to train a machine learning algorithm to predict user's emotions. Experiments indicate that J48 was the best machine learning algorithm for this approach.

IV. SYSTEM ARCHITECTURE AND IMPLEMENTATION

Figure 4 shows the basic system architecture. As the Figure shows, there are two types of users: an Application User and a Web User. An Application User interacts with the mobile phone while the Web User uses the browser to view public trends.

As Figure 4 shows, the core of the project is the soft keyboard application running on the Android phone. This Keyboard Application is responsible for collecting sensor data and applying machine learning techniques to predict the current emotion of the learner. J48 Classifier from Weka's library was modified to run on the Android-based devices. The ported library supports any mobile phone supporting Android V2.0 (Eclair) or higher. The Keyboard Application stores feature vectors in the form of the log file in WEKA's ARFF format. This file is used as input to train the J48 Classifier. As Figure 4 shows, the soft keyboard also publishes user's emotional state into a web application. This is done through RESTful API. The Web Server is implemented using Python and generates geographical and other charts that depict the average emotional state of countries with registered users. Personalized charts depicting an individual's emotional state can only be accessed by the relevant user. The web-application uses the PostgreSQL DB to store all the data obtained from users.

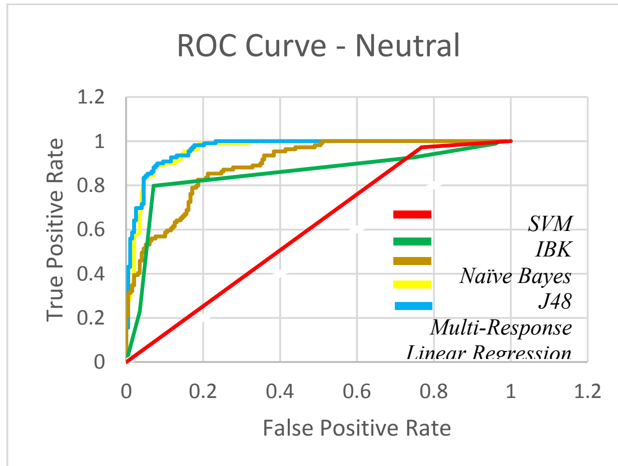

Fig. 1. ROC Curve for "Neutral" State.

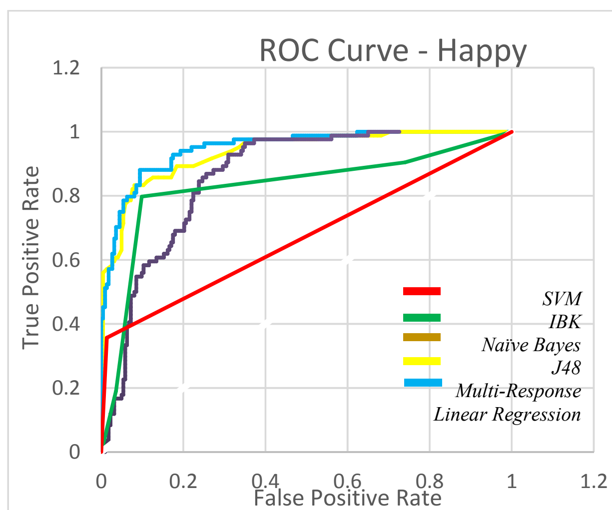

Fig. 2. ROC Curve for "Happy" State.

TABLE 1. RESULTS FOR J48.

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
	0.821	0.067	0.821	0.821	0.821	0.918	Happy
	0.872	0.051	0.905	0.872	0.888	0.94	Neutral
	1	0	1	1	1	1	Angry
	0.979	0.019	0.904	0.979	0.94	0.987	Sad
Avg.	0.902	0.039	0.902	0.902	0.902	0.954	

TABLE 2. RESULTS FOR SVM.

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
	0.357	0.013	0.909	0.357	0.513	0.672	Happy
	0.972	0.768	0.411	0.972	0.578	0.602	Neutral
	0.076	0	1	0.076	0.141	0.538	Angry
	0.229	0	1	0.229	0.373	0.615	Sad
Avg.	0.495	0.276	0.766	0.495	0.434	0.609	

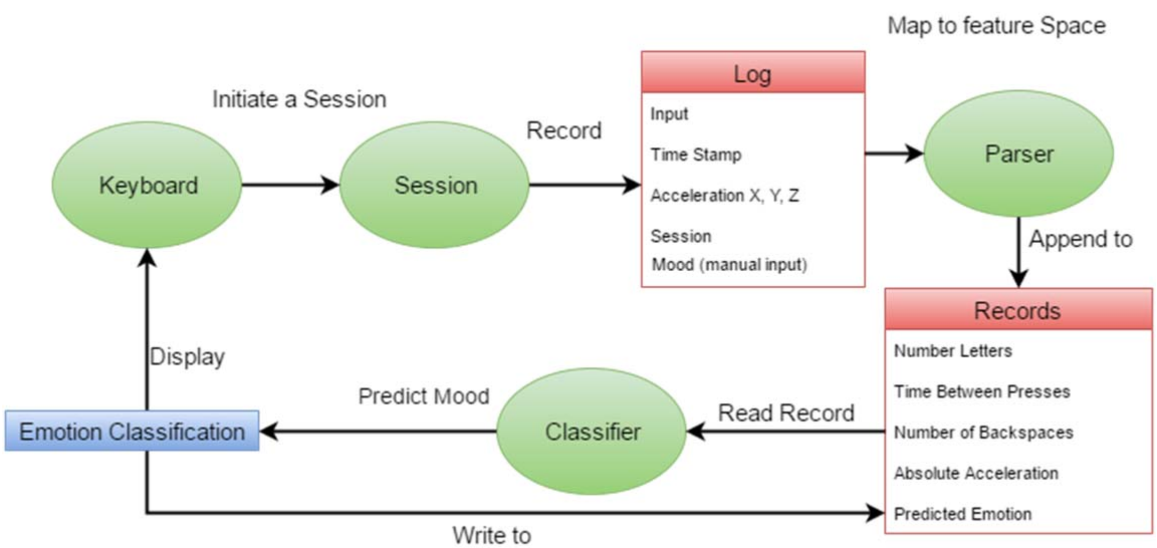

Fig. 6. Data Capture and Prediction.

The Keyboard Application and the Web Server are described in more detail next.

A. Keyboard Application

The Keyboard Application was developed by modifying the Android Open Source Project (AOSP) keyboard [22] to allow it to capture data while the user is typing. The response time of the keyboard was also improved to allow for a smoother user experience. The keyboard captures the time between key presses in milliseconds and the number of backspaces as a measure of mistakes made in input by the user. This is all captured periodically during a period of 5 seconds, wherein each period is denoted as a segment.

In the first phase, the algorithm requires the user to enter his current emotion in order to construct the training set which consists of pre-classified data. This is done through utilizing the emotion bar on top of the keyboard as seen in Figure 5. It has been empirically determined that the algorithm requires at least 150 segments to start predicting the user's emotional state within a reasonable margin of error.

After the training phase is over, the keyboard data is used to predict the current emotion of the user using the J48 algorithm. As Figure 6 shows, when used, the keyboard initiates a session lasts five seconds. Within a session, the keyboard logs user typing behavior, acceleration in three dimensions and the emotion recorded from the candidate bar. This data is stored in a log file which is then processed to create a feature vector in the feature space which is appended to the Record file. At the end of every session, the algorithm is re-trained based on all previous entries in the Record file and the latest entry is passed through the J48 algorithm to be classified. The classified entry is displayed on the candidate bar of the keyboard and is added to the emotion attribute of the last entry in the file.

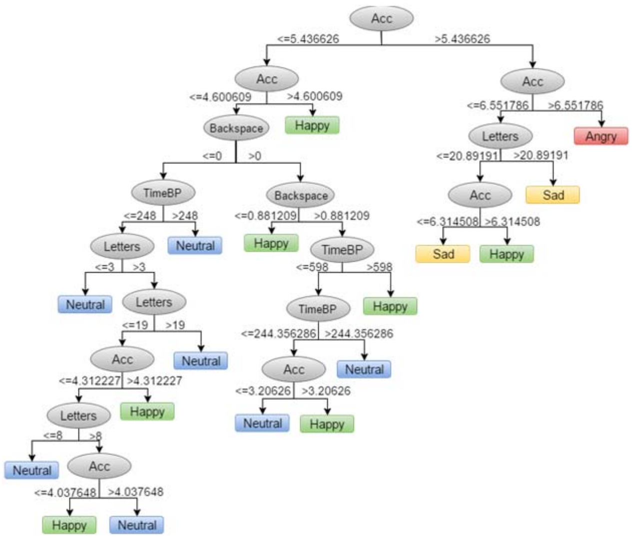

Fig. 3. Decision Tree Generated by J48. "TimeBP" indicates Average Time between Key Strokes in milliseconds.

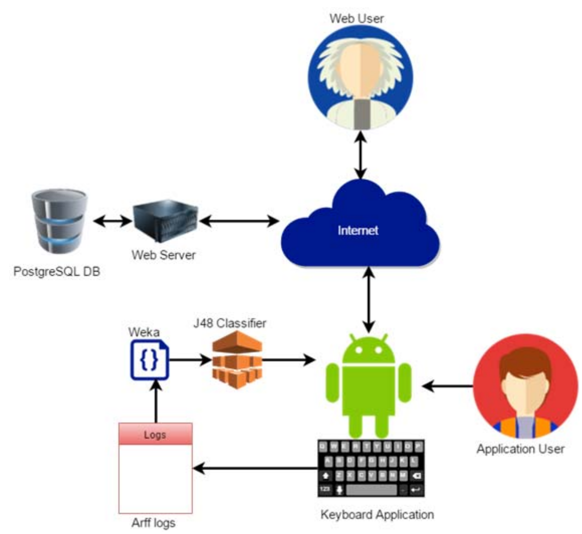

Fig. 4. Basic System Architecture.

Fig. 5: Emotion Bar Options.

Figure 7 shows how depending on user's typing behavior, the current emotive state is changed from neutral to angry.

When the user first installs the keyboard, they are greeted with a log-in screen which allows them to log-in or to create a new account. Since the keyboard will not keep track of the user's emotions if the user is not logged in, the log in screen keeps popping up until the user logs in. When the user logs-in, the keyboard receives a token which is then stored on the phone for later use. After logging in, at certain time intervals, the current emotion of the user is sent to the Web Server along with a timestamp and the geolocation of the user which is found based on the user's service provider. If the user does not have a service provider, he or she is given the option to manually change their location from the keyboard's settings menu. This eliminates the need for the use of GPS and location services. The emotion, timestamp, and country are then sent to the server via a HTTP POST request. The token received earlier when the user logged-in is placed as a header for the POST request, with the emotion, timestamp, and country placed in the body of the request.

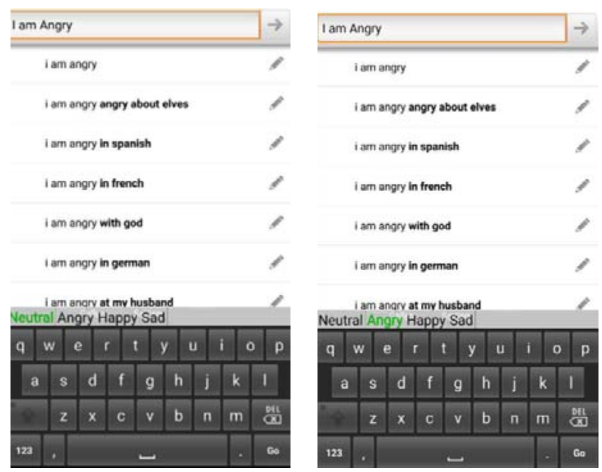

Fig. 7. User's predicted state changing based on their input.

B. *Web Server Application*

The Web Server is implemented in Python using a web framework called Django (*http://www.djangoproject.com/*). At the core of the WebServer is the PostgreSQL database that stores the information about the users registered to the website, and the information about their emotional states.

The browser-based User Interface (UI) for the Web Server was designed by using the bootstrap3 framework (*http://getbootstrap.com/*) which allows for rapid integration of common, responsive, designs. The website's navigation is consistent across the entire website in order to maintain a consistent experience. Furthermore, Django's messaging middleware was used to allow for responsive messages that inform the user of what is going on as he or she interacts with the Web Server.

To provide maps that depict the emotional states across regions, Google's GeoCharts (*http://developers.google.com/chart/interactive/docs/gallery/geochart*) was used to generate a choropleth that illustrates the modal emotional state across the region.

V. Conclusions and Future Work

This paper presented a machine learning approach for emotion recognition using a mobile phone soft keyboard. The keyboard records the user's typing behavior that includes texting speed and time between presses, and shaking as measured through the built-in accelerometer. The keyboard dynamically uses the J48 machine learning algorithm in order to classify the user's current mood. The system also sends anonymized user data to a server that can be publicly accessed to view demographic information. The demographic data could be used by researchers in various fields and disciplines. The system demonstrates that it is possible to enable emotion recognition on mobile phones using built-in sensors. The system also does so in an application independent manner where any mobile application using a keyboard for input can use the proposed service. The emotion detection service can be used in many domains, such as healthcare, social media, trading, etc.

The current work has limitations and can be improved in multiple ways. First, classification accuracy, though high, can potentially be improved by incorporating additional attributes. For example, location (home, work, etc.), time, intensity of finger strokes, usage of strong language, facial expression, ambient temperature, weather data and discomfort index can be considered. These additional parameters can potentially help improve the accuracy rate and allow one to add more emotional states to the list of emotion the application can detect. Secondly, the keyboard layout can be made more appealing to the user as well. Finally, on the Web Server side, third-party login services can be incorporated to allow users to quickly sign up to the service.

References

[1] F. Aloul, I. Zualkernan, R. Abu-Salma, H. Al-Ali, and M. Al-Merri, "iBump: Smartphone Application to Detect Car Accidents," in *Proc. of the IEEE International Conference on Industrial Automation and Information & Communications Technology* (IAICT), pp. 52-56, August 2014.

[2] M. Al-Mardini, F. Aloul, A. Sagahyroon, and L. Al-Husseini, "Classifying Obstructive Sleep Apnea Using Smartphones," *Elsevier Journal of Biomedical Informatics*, vol. 52, pp. 251–259, December 2014.

[3] H. Hasan, B. Safieh, F. Aloul, and A. Sagahyroon, "Diagnosing COPD Using Mobile Phones," in *Proc. of the IEEE International Conference on Electronics, Information and Communication* (ICEIC), January 2015.

[4] K. Button, G. Lewis, and M. Munafò, "Understanding emotion: Lessons from anxiety," *Behavioral and Brain Sciences*, vol. 35, no. 3, pp. 145, June 2012.

[5] M. Munezero, C. Montero, E. Sutinen, and J. Pajunen, "Are They Different? Affect, Feeling, Emotion, Sentiment, and Opinion Detection in Text," *IEEE Transactions on Affective Computing*, vol. 5, no. 2, pp. 101-111, April 2014.

[6] A. Agrawal and A. An, "Unsupervised Emotion Detection from Text Using Semantic and Syntactic Relations," in *Proc. of the IEEE/WIC/ACM International Joint Conferences on Web Intelligence and Intelligent Agent Technology* (WI-IAT), pp. 346-353, 2012.

[7] B. Shenghua, S. Xu, L. Zhang, R. Yan, Z. Su, D. Han, and Y. Yu, "Mining Social Emotions from Affective Text," *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, no. 9, pp. 1658-1670, September 2012.

[8] F. Zubaydi, A. Saleh, F. Aloul, and A. Sagahyroon, "Security of Mobile Health (mHealth) Systems," in *Proc. of the IEEE International Conference on Bioinformatics and Bioengineering* (BIBE), 1-5, November 2015.

[9] S. Shivhare and S. Saritha, "Emotion Detection From Text Documents," *International Journal of Data Mining & Knowledge Management Process*, vol. 4, no. 6, pp. 51-57, November 2014.

[10] C. Lu, W. William, H. Peng, J. Chung, and J. Ho, "Emotion Sensing for Internet Chatting: A Web Mining Approach for Affective Categorization of Events," in *Proc. of the International Conference on Computational Science and Engineering* (CSE), pp. 295-301, December 2010.

[11] G. Sandstrom, J. Rentfrow, and C. Mascolo, "EmotionSense," *University of Cambridge and University of Essex*, Available at: *http://emotionsense.org/*

[12] "T2 Mood Tracker," *The National Center for Telehealth & Technology*, 2012. Available at: *http://t2health.dcoe.mil/*

[13] F. Sebastiani, "Machine Learning in Automated Text Categorization," *ACM Computing Surveys*, vol. 34, no. 1, pp. 1-47, 2002.

[14] A. McCallum and K. Nigam, "A Comparison of Event Models for Naive Bayes Text Classification," in AAAI/ICML *Workshop on Learning for Text Categorization*, pp. 41-48, 1998.

[15] N. Sebe, M. Lew, I. Cohen, A. Garg, and T. Huang, "Emotion recognition using a Cauchy Naive Bayes classifier," in *International Conference on Pattern Recognition*, pp.17-20, 2002.

[16] S. Ali, S. Zehra, and A. Arif, "Performance Evaluation of Learning Classifiers for Speech Emotions Corpus using Combinations of Prosodic Features," *International Journal of Computer Applications*, vol. 76, no. 2, pp. 35-43, 2013.

[17] C. Nie, J. Wang, F. He, and R. Sato, "Application of J48 Decision Tree Classifier in Emotion Recognition Based on Chaos Characteristics," in *Proc. of the International Conference on Automation, Mechanical Control and Computational Engineering*, pp. 1847-1850, April 2015.

[18] R. Duda, P. Hart, and D. Stork, "Pattern Classification," *Wiley-Interscience*, 2nd Edition, pp. 174-184, 2000.

[19] I. Witten and E. Frank, "Data Mining: Practical Machine Learning Tools and Techniques with Java implementations," *Morgan Kaufmann Series in Data Management Systems,* 2nd edition, 2005.

[20] M. Dumas, "Emotional Expression Recognition using Support Vector Machines," *Technical Report*, University of California, San Diego, 2001.

[21] Weka 3. Available at: *http://www.cs.waikato.ac.nz/ml/weka/*

[22] Android Open Source Project. Available at: *http://source.android.com/*